



เอกสารอ้างอิง

1. Paul M. Anderson, Analysis Of Faulted Power System, The Iowa State University Press / Ames, 1976
2. Glenn W. Stagg, and Ahmed H. El-Abiad, Computer Methods In Power System Analysis, McGraw-hill, 1985
3. William D. Stevenson, Jr. , Elements of Power System Analysis, McGraw-Hill, 1982
4. A. Brameller, M.N. John, Mr. Scott, Practical Diakoptics For Electrical Networks, Chapman & Hall Ltd., 1969
5. H.H. Happ, Piecewise Methods And Application To Power Systems, John Wiley & Sons, 1980
6. H.H. Happ, And Undrill J.M., "Automatic Sectionalization Of Power System Networks For Network Solution", IEEE Trans. Power Apparatus And System 90 (January/February 1971) : 46-53
7. H.H Happ, and C.C Young, "Tearing Algorithms For Large Scale Network Programs", IEEE Trans. Power Apparatus And System 90 (November/December 1971) : 2639-2649
8. J.M.Undrill, and T.E.Kostyniak, "Advanced Power System Fault Analysis Method", IEEE Trans. Power Apparatus And Systems 90 ,Vol.PAS-94,no.6, (November/December 1975) :2141-2150
9. H.A.Daniels, and M.-S.Chen, "Compensation of Mutual Coupling in the Formation of the Bus Impedance Matrix", IEEE Trans.Power Apparatus And System 90 (March 1971)
10. Charles A. Gross, Power System Analysis, John Wiley & Sons, 1986
11. Turan Goren, Modern Power System Analysis, John Wiley & Sons, 1988



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ภาคผนวก ก.
คู่มือการใช้โปรแกรม

โปรแกรมวิเคราะห์การลัดวงจรและโปรแกรมวิเคราะห์การลัดวงจรโดยวิธีแยกเป็นข้อ
FAULT ANALYSIS (FA) AND FAULT ANALYSIS BY PIECEWISE SOLUTION METHOD (FP)

โปรแกรม FA และ FP เป็นโปรแกรมสำหรับวิเคราะห์การลัดวงจรของระบบไฟฟ้ากำลัง สามารถใช้ในการวิเคราะห์ระบบไฟฟ้าเพื่อเป็นข้อมูลในการออกแบบระบบป้องกัน (PROTECTION SYSTEM) ต่างๆ โปรแกรมทั้งสองมีข้อแตกต่างกันตรงดังนี้

- โปรแกรม FA จะสร้างบัสสมิพินเดนซ์เมตริกซ์ของทั้งระบบโดยใช้วิธีอัลกอริทึม
- โปรแกรม FP จะแบ่งระบบทั้งหมดออกเป็นสองส่วน สร้างเมตริกซ์เชื่อมโยงระบบย่อย (Z_1, Z_2, Z_4) แล้วจึงสร้างบัสสมิพินเดนซ์เมตริกซ์เฉพาะคอลัมน์ของบัสที่ต้องการวิเคราะห์

โปรแกรมทั้งสองจะทำงานในลักษณะโปรแกรม โหลด คือป้อนข้อมูลลงในตารางสำเร็จรูป ที่เตรียมไว้ ง่ายต่อการใช้งาน ตั้งแต่การเตรียมข้อมูลการป้อนข้อมูล การแก้ไขข้อมูล การคำนวณ และการแสดงผล

โปรแกรมเขียนด้วยภาษาซี (C Language) สำหรับไมโครคอมพิวเตอร์ของไอบีเอ็ม (IBM) ขนาด 16 บิต (IBM PC, XT หรือ AT) คอมไพล์ด้วยซีคอมไพเลอร์ (C Compiler) สามารถเรียกใช้ได้โดยตรงภายใต้ดอส (DOS) โปรแกรมทั้งหมดแบ่งออกเป็นโปรแกรมย่อย ๆ ดังนี้

1. FA.EXE เป็นโปรแกรมวิเคราะห์การลัดวงจร
2. FP.EXE เป็นโปรแกรมวิเคราะห์การลัดวงจรโดยวิธีแยกเป็นส่วนย่อย

โปรแกรมทั้งสองจะอยู่ในดิสก์เดียวกัน โดยที่ไฟล์ต่าง ๆ ที่เกี่ยวกับข้อมูลอาจจะอยู่ในดิสก์เดียวกันกับโปรแกรม

ขนาดของ ไมโครคอมพิวเตอร์ และส่วนประกอบ ควรมีดังนี้

1. ไมโครคอมพิวเตอร์ของ IBM (PC, XT หรือ AT) ซึ่งมีหน่วยความจำอย่างน้อย 256 K
2. ฟลอปปีดิสก์ไดรฟ์อย่างน้อย 1 ตัว
3. PC-DOS operating system
4. ดอทเมตริกซ์พริ้นเตอร์ของ EPSON (Dot matrix printer)

1. การเริ่มต้นใช้โปรแกรม

โปรแกรม FA และ FP จะทำงานภายใต้ดอส เมื่อไมโครคอมพิวเตอร์ทำงานอยู่ภายใต้ดอส การเริ่มต้นของโปรแกรมทำได้ดังนี้

1. พิมพ์คำว่า FA หรือ FP แล้วกด ENTER โปรแกรม FA.EXE หรือ FP.EXE จะถูกโหลดเข้าเครื่อง และจะปรากฏ Frame Work ดังรูปที่ 1

Cell: 1A

File: S

JOBS

OldData:

Date: 06-14-62

Require: NUMBER; 0 to 49

Time: 06:00:00

Line	Bus		Equip. Type	Self Impedance		Mutual Impedance	
	P	Q		Positive	Zero	Zm	Line
1							
2							
3							
4							
5							
6							
7							
8							
9							
10							
11							
12							
13							
14							
15							

F2-Save

F6-Print

F10,-/MEMU

รูปที่ 1

ศูนย์วิทยทรัพยากร

ดังแสดงในรูปที่ 1 Frame Work สามารถแบ่งออกได้เป็น 3 ส่วนดังนี้

ส่วนที่ 1 ด้านบนสุดหรือสามบรรทัดแรกของจอภาพ คือส่วนที่แสดงสถานะของโปรแกรมซึ่งจะกล่าวถึงในภายหลัง

ส่วนที่ 2 ส่วนที่อยู่ในตาราง เป็นข้อมูลที่ใช้ในการคำนวณ แบ่งออกเป็นข้อมูลชนิดต่าง ๆ ดังนี้

- หมายเลขลายน์
- หมายเลขบัสประจำลายน์ (บัส P และ Q)
- ชนิดของลายน์ (สายส่ง, หม้อแปลงหรือเครื่องกำเนิดไฟฟ้า)
- ลักษณะของลายน์ เช่นเป็นหม้อแปลงชนิด YG-YG
- ค่าอิมพีแดนซ์ในซีเควนซ์บวกรของลายน์

- ค่าอิมพีแดนซ์ในซีเควนซ์ศูนย์ของลายน
- ค่าอิมพีแดนซ์ในซีเควนซ์ศูนย์ของลายนที่มิวชวล (Mutual) กัน
- หมายเลขลายนที่มิวชวลกัน

ส่วนที่ 3 โมดการทำงานของโปรแกรมในส่วนที่ไม่เกี่ยวกับการคำนวณซึ่งจะกล่าวถึงในหัวข้อที่ 11

2. กด " F 10 " หรือ " / " เพื่อเลือกโมดการทำงานของโปรแกรม ซึ่งจะปรากฏดังรูปที่ 2

Cell: 1A File: S MENU

Worksheet	File	Calculate	Display	Print	Exit
Line	Bus ----- P Q	Equip. Type	Self Impedance ----- Positive Zero	Mutual ----- Zm	No Yes e -- ne
1					
2					
3					
4					
5					
6					
7					
8					
9					
10					
11					
12					
13					
14					
15					

F6-Print ←-Do command Esc-Go back

ศูนย์วิทยุทรัพยากร

รูปที่ 2

จุฬาลงกรณ์มหาวิทยาลัย

ดังแสดงในรูปที่ 2 บรรทัดบนเหนือตารางจะปรากฏโมดการทำงานของโปรแกรม ซึ่งสามารถเลือกได้ โดยการเลื่อนตัวชี้เซลล์ไปยังโมดที่ต้องการ โดยการกดปุ่มที่มีสัญลักษณ์ " " = ไปซ้าย " " = ไปขวา พร้อมทั้งกด ENTER เมื่อต้องการเลือกโมด

2. Work Sheet

Cell: 1A

File: S

MENU

Worksheet	File	Calculate	Display	Print	Exit
Set Date	uip. Type	Self Impedance		Mutual Impedance	
Set Time		-----		-----	
Clear table		Positive	Zero	Zm	Line
1					
2					
3					
4					
5					
6					
7					
8					
9					
10					
11					

รูปที่ 3
 ดังแสดงในรูปที่ 3 สามารถเลือกโหมดย่อยได้ โดยกดปุ่มที่มีสัญลักษณ์ " " = ขึ้น,
 " " = ลง พร้อมทั้งกด ENTER เพื่อเลือกโหมดย่อยต่าง ๆ ดังนี้

2.1 Set Date จะปรากฏข้อความดังนี้
 Enter Date (MM-DD-YY) :

ผู้ใช้เพียงป้อนเดือน วันที่ ปี ที่ต้องการ และกด ENTER วันเวลาที่ระบุจะปรากฏบน
 Frame Work ในบรรทัดที่สองด้านขวา

2.2 Set Time จะปรากฏข้อความดังนี้
 Enter Time (HH:MM:SS)

ผู้ใช้เพียงป้อนเวลาที่ต้องการและกด ENTER เวลาที่ระบุจะปรากฏบน Frame Work ใน
 บรรทัดที่สามด้านขวา

2.3 Clear Table เป็นการลบข้อมูลในตารางออกทั้งหมด เพื่อรอการป้อนข้อมูลใหม่

3. File

Cell: 1A

File: S

MENU

Worksheet	File	Calculate	Display	Print	Exit
Line	Bus ----- P Q	Load Write to	Self Impedance ----- Positive Zero		Mutual Impedance ----- Zm Line
1					
2					
3					
4					
5					
6					
7					
8					

รูปที่ 4

ดังแสดงในรูปที่ 4 สามารถเลือก โมดย่อยต่าง ๆ ได้ดังนี้

3.1 Load จะปรากฏข้อความดังนี้
ENTER file name for loading

ผู้ใช้เพียงป้อนชื่อ ไฟล์ที่ต้องการวิเคราะห์ และกด ENTER เครื่องจะ โหลด ไฟล์ดังกล่าวลงในตารางพร้อมทั้งปรากฏชื่อไฟล์ในบรรทัดแรกของ Frame Work ถ้าชื่อ ไฟล์ไม่มีอยู่ในดิสค์จะปรากฏข้อความบนด้านล่างของตารางดังนี้

" Error " File not found

ให้กดปุ่มใด ๆ ก็ได้ 1 ครั้ง (หมายถึงตัวอักษร A,B,C,.....,TAB ฯลฯ ที่ไม่ใช่ปุ่มฟังก์ชันหรือปุ่มกดควบคุมต่าง ๆ) โปรแกรมจะกลับเข้าสู่โหมดพร้อม (Ready Mode) ซึ่งคือ Frame Work ตามรูปที่ 1 ให้กดปุ่ม " F 10 " หรือ " / " เพื่อกลับเข้าสู่โหมดการทำงานของโปรแกรมอีกครั้ง

3.2 Write to จะปรากฏข้อความดังนี้
Enter new file name

ผู้ใช้เพียงป้อนชื่อ ไฟล์ที่ต้องการให้เก็บข้อมูลที่ปรากฏบนตาราง แล้วกด ENTER เครื่องจะโหลดข้อมูลดังกล่าว ไปเก็บไว้ในชื่อ ไฟล์ที่ต้องการ

4. Calculation

Cell: 1A

File: S3

MENU

Worksheet	File	Calculate	Display	Print	Exit
Line	Bus ----- Equip. Type P Q	Type of Fault : 3 Phase Referece Phase : A Fault at Bus : 1 Quantities : Sequence Type of Solution : Partial Fault Calculation	Mutual Impedance ----- Zm Line		
1	0 1 GEN YG				
2	0 3 GEN YG				
3	0 5 GEN YG				
4	0 7 GEN Y	0.0020 0.1000	0.0010	0.0800	
5	1 7 LINE	0.0670 0.2000	0.1350	0.6500	

รูปที่ 5

ดังแสดงในรูปที่ 5 เป็นโมดการทำงานที่ใช้สำหรับกำหนดชนิด และลักษณะของการลัดวงจร รวมถึงบัลต์ที่ต้องการวิเคราะห์ โดยสามารถเลือกโมดย่อยต่าง ๆ ได้ดังนี้

4.1 Type of Faults: เป็นโมดย่อยที่กำหนดชนิดของการเกิดลัดวงจร โปรแกรม FA จะวิเคราะห์การลัดวงจร 4 ชนิด โดยที่จอภาพจะปรากฏข้อความดังรูปที่ 5.1

Cell: 1A

File: S3

MENU

Worksheet	File	Calculate	Display	Print	Exit
Line	Bus ----- Equip. Type P Q	Type of Fault : 3 Phase Referece Phase : A Fault at Bus : 1 Quantities : Sequence Type of Solution : Partial Fault Calculation	Mutual Impedance ----- Zm Line		
1	0 1 GEN YG				
2	0 3 GEN YG				
3	0 5 GEN YG				
4	0 7 GEN Y			0.0800	
5	1 7 LINE	Enter Fault No. :		0.6500	
6	1 9 LINE			0.6000	
7	1 2 XMER D-YG				
8	2 11 LINE	1. Three Phase (3 Phase)			
9	3 5 LINE	2. Line to Line (LL)			
10	3 10 LINE	3. Single Line to Ground (SLG)			
11	3 10 LINE	4. Double Line to Ground (2LG)	0.0000	0.2000	10
12	3 4 XMER D-YG				
13	4 12 LINE	0.3500 0.4200	0.7000	1.2000	
14	5 14 LINE	0.0670 0.2000	0.1500	0.6500	
15	5 14 LINE	0.0670 0.2000	0.1750	0.7000	0.0000 0.2200 14

F6-Print

←-Inout

Esc-Go back

รูปที่ 5.1

ผู้ใช้เพียงป้อนหมายเลขของชนิดของการลัดวงจรที่ต้องการวิเคราะห์ แล้วกด ENTER จะปรากฏชนิดของการลัดวงจรต่อจากโมดย่อย

ในกรณีที่ใช้โปรแกรม FP จะวิเคราะห์การลัดวงจรเพียง 2 ชนิด โดยที่จอภาพจะปรากฏ

ข้อความดังรูปที่ 5.2

Cell: 1A File: S3 MENU

Worksheet	File	Calculate	Display	Print	Exit
Line	Bus ---- Equip. Type P Q	Type of Fault : 3 Phase Reference Phase: A Fault at Bus : 1 Quantities : Sequence Fault Calculation	Mutual Impedance ----- Zm Line		
1	0 1 GEN YG 0.0				
2	0 3 GEN YG 0.0				
3	0 5 GEN YG 0.0				
4	0 7 GEN Y 0.0	Enter Type No. : 0500			
5	1 7 LINE 0.0				
6	1 9 LINE 0.0				
7	1 2 XMER D-YG 0.0	1. Three Phase (3 Phase)			
8	2 11 LINE 0.3	2. Line to Line (LL)			
9	3 5 LINE 0.0				
10	3 10 LINE 0.0670	0.2000 0.1500 0.7000			
11	3 10 LINE 0.0670	0.2000 0.1500 0.7000	0.0000 0.2000 10		

รูปที่ 5.2

ซึ่งวิธีการกำหนดชนิดของการลัดวงจรจะเหมือนกับโปรแกรม FA

4.2. Reference Phase : เป็นโมดย่อยที่กำหนดเฟสอ้างอิงของการวิเคราะห์

การลัดวงจรชนิดต่าง ๆ โดยที่จอภาพจะปรากฏข้อความดังรูปที่ 5.3

Cell: 1A File: S3 MENU

Worksheet	File	Calculate	Display	Print	Exit
Line	Bus ---- Equip. Type P Q	Type of Fault : 3 Phase Reference Phase : A Fault at Bus : 1 Quantities : Sequence Type of Solution : Partial Fault Calculation	Mutual Impedance ----- Zm Line		
1	0 1 GEN YG				
2	0 3 GEN YG				
3	0 5 GEN YG				
4	0 7 GEN Y				
5	1 7 LINE	Enter Phase (A, B or C) :			
6	1 9 LINE				
7	1 2 XMER D-YG	0.0000 0.1200 0.0000 0.1200			
8	2 11 LINE	0.3500 0.4200 0.1340 0.6000			

รูปที่ 5.3

ผู้ใช้เพียงป้อนชื่อเฟสอ้างอิงที่ต้องการวิเคราะห์ แล้วกด ENTER จะปรากฏชื่อเฟสอ้างอิงของการวิเคราะห์ต่อจากโมดย่อย

4.3 Fault at Bus : เป็นโมดย่อยที่กำหนดตำแหน่งของบัสที่ต้องการวิเคราะห์ โดยที่จอภาพจะปรากฏข้อความดังรูปที่ 5.4

Cell: 1A File: S3 MENU

Worksheet	File	Calculate	Display	Print	Exit
Line	Bus ----- P Q	Equip. Type	Type of Fault : 3 Phase Referece Phase : A Fault at Bus : 1 Quantities : Sequence Type of Solution : Partial Fault Calculation	Mutual Impedance ----- Zm Line	
1	0 1	GEN YG			
2	0 3	GEN YG			
3	0 5	GEN YG			
4	0 7	GEN Y			
5	1 7	LINE	Enter Bus No. (0 for ALL) :		
6	1 9	LINE			
7	1 2	XMER D-YG	0.0000 0.1200 0.0000 0.1200		
8	2 11	LINE	0.3500 0.4200 0.1340 0.6000		

รูปที่ 5.4

ผู้ใช้เพียงป้อนหมายเลขของบัสที่ต้องการวิเคราะห์ แล้วกด ENTER หรือในกรณีที่ต้องการวิเคราะห์ทุกบัส ผู้ใช้เพียงป้อนหมายเลข 0 เท่านั้น

4.4 Quantities : เป็นโมดย่อยที่กำหนดชนิดของผลลัพธ์ว่าเป็นค่าบนแกนรีเดนท์หรือค่าบนแกนเฟส โดยที่จอภาพจะปรากฏข้อความดังรูปที่ 5.5

จุฬาลงกรณ์มหาวิทยาลัย

Cell: 1A

File: S3

MENU

Worksheet	File	Calculate	Display	Print	Exit
Line	Bus ----- P Q	Equip. Type	Type of Fault : 3 Phase Referece Phase : A Fault at Bus : 1 Quantities : Sequence Type of Solution : Partial Fault Calculation	Mutual Impedance ----- Zm Line	
1	0 1	GEN YG			
2	0 3	GEN YG			
3	0 5	GEN YG			
4	0 7	GEN Y			
5	1 7	LINE	Enter Quantity No. : 0800		
6	1 9	LINE	6500		
7	1 2	XMER D-YG	6000		
8	2 11	LINE	1. Sequence (0, 1, 2) .1200		
9	3 5	LINE	2. Phase (A, B, C) .6000		
10	3 10	LINE	0. Sequence & Phase .7000		
11	3 10	LINE	.7000	0.0000	0.2000 10

รูปที่ 5.5

ผู้ใช้เพียงป้อนหมายเลข 1, 2 หรือ 3 แล้วกด ENTER

4.5 Solution : เป็นโมดูลย่อยที่กำหนดการแสดงผลการคำนวณ ว่าต้องการให้แสดงผลเฉพาะบัส/สายน์ ที่ต่อกับบัสที่เกิดลัดวงจร หรือแสดงผลของทุกบัส/สายน์ของระบบไฟฟ้า กำลังที่กำลังวิเคราะห์อยู่ โดยที่จอภาพจะปรากฏข้อความดังรูปที่ 5.6

Cell: 1A

File: S3

MENU

Worksheet	File	Calculate	Display	Print	Exit
Line	Bus ----- P Q	Equip. Type	Type of Fault : 3 Phase Referece Phase : A Fault at Bus : 1 Quantities : Sequence Type of Solution : Partial Fault Calculation	Mutual Impedance ----- Zm Line	
1	0 1	GEN YG			
2	0 3	GEN YG			
3	0 5	GEN YG			
4	0 7	GEN Y			
5	1 7	LINE	Enter Solution No. : 0800		
6	1 9	LINE	6500		
7	1 2	XMER D-YG	0.0000	0.1200	
8	2 11	LINE	1. Partial	0.1340	0.6000
9	3 5	LINE	2. All	0.1700	0.6000
10	3 10	LINE		0.1500	0.7000
11	3 10	LINE	0.0670	0.2000	0.1500 0.7000
12	3 4	XMER D-YG	0.0000	0.1200	0.0100 0.3000

รูปที่ 5.6

เพียงผู้ใช้ป้อนหมายเลข 1 หรือ 2 แล้วกด ENTER

4.6 Fault Calculating เป็นโมดย่อยที่กำหนดให้โปรแกรมเริ่มคำนวณเพียงผู้ใช้กด ENTER เท่านั้น จะปรากฏผลลัพธ์บนจอภาพดังรูปที่ 6

BALANCE THREE PHASE FAULT AT BUS 1 REFERENCE PHASE : A						
SEQUENCE QUANTITIES						
BUS CURRENT						
Bus		I0	ANGO	I1	ANG1	I2 ANG2
1		0.0000	0.00	105.9031	-89.05	0.0000 0.00
BUS VOLTAGE						
Bus		V0	ANGO	V1	ANG1	V2 ANG2
1		0.0000	0.00	0.0000	0.00	0.0000 0.00
2		0.0000	0.00	0.0649	27.93	0.0000 0.00
7		0.0000	0.00	0.6958	-5.02	0.0000 0.00
9		0.0000	0.00	0.4892	-0.33	0.0000 0.00
LINE CURRENT						
Bus to bus		I0	ANGO	I1	ANG1	I2 ANG2
0 1		0.0000	0.00	100.0000	-90.00	0.0000 0.00
7 1		0.0000	0.00	3.2990	-76.50	0.0000 0.00
9 1		0.0000	0.00	2.3191	-71.81	0.0000 0.00
2 1		0.0000	0.00	0.5407	-62.07	0.0000 0.00

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

รูปที่ 6

5. Display

Cell: 1A

File: S2

MENU

Worksheet	File	Calculate	Display	Print	Exit
Line	Bus ----- P Q	Equip. Type	Self I ----- Positive	Data for Positive Seq. Z-BUS for Positive Seq. Data for Zero Seq. Z-BUS for Zero Seq. Fault Solution	Impedance ----- Line
1	0 1	GEN Y	0.0200 0.2400		
2	0 2	GEN YG	0.0300 0.4800		
3	1 4	LINE	0.1600 0.7400	0.8000 1.8500	
4	1 6	LINE	0.2460 1.0360	0.9840 2.0840	
5	2 3	LINE	1.4460 2.1000	3.7800 5.2600	
6	2 5	LINE	0.5640 1.2800	2.8200 3.8400	

รูปที่ 7

ดังแสดงในรูปที่ 7 เป็นโมดการทำงานที่ใช้สำหรับแสดงผลการวิเคราะห์ต่าง ๆ ลงบนจอภาพของโปรแกรม FA โดยสามารถเลือกโมดย่อยต่าง ๆ ได้ดังนี้

5.1 Data for Positive Seq. เป็นโมดย่อยที่แสดงค่าอิมพีแดนซ์บวกของสายนี้ต่าง ๆ ของระบบไฟฟ้ากำลัง ก่อนการคำนวณ เพียงผู้ใช้กด ENTER เท่านั้น จะปรากฏภาพดังรูปที่ 8

DATA FOR POSITIVE SEQUENCE					
Number of Lines = 9			Number of Buses = 6		
LINE	BUS	BUS	POSITIVE		
1	0	1	0.0200	+0.2400j	
2	0	2	0.0300	+0.4800j	
3	1	4	0.1600	+0.7400j	
4	1	6	0.2460	+1.0360j	
5	2	3	1.4460	+2.1000j	
6	2	5	0.5640	+1.2800j	
7	3	4	0.0000	+0.2660j	
8	6	4	0.1940	+0.8140j	
9	5	6	0.0000	+0.6000j	

รูปที่ 8

ให้กดปุ่มใด ๆ ก็ได้ 1 ครั้ง การทำงานจะกลับสู่โมด Display ดังรูปที่ 7

5.2 Z-BUS for Positive Seq. เป็นโมดย่อยที่แสดงค่าบัสโวลิตจเป็นแดนซ์เมตริกซ์ที่เควนซ์
 บวกของระบบไฟฟ้ากำลัง เพียงผู้ใช้กด ENTER เท่านั้น จะปรากฏภาพดังรูปที่ 9

ZBUS FOR POSITIVE SEQUENCE			
BUS	BUS	R	X
1	1	0.02253	0.21503
1	2	-0.00609	0.04974
1	3	0.02635	0.16117
1	4	0.02254	0.17266
1	5	0.02118	0.12825
1	6	0.01831	0.16379
2	2	0.04422	0.38094
2	3	-0.01594	0.15713
2	4	-0.00786	0.13434
2	5	-0.00698	0.22306
2	6	0.00023	0.15223
3	3	0.16244	0.73912
3	4	0.14333	0.53368
3	5	0.06192	0.27007
3	6	0.07295	0.32786
4	4	0.13269	0.57694
4	5	0.06506	0.27818
4	6	0.06881	0.34726
5	5	0.16569	0.80649
5	6	0.13256	0.46538
6	6	0.13034	0.61119

Time used = 3.5 Seconds
 Memory used = 36 Units

จุฬาลงกรณ์มหาวิทยาลัย

5.3 Data for Zero Seq. เป็นโมดย่อยที่แสดงค่าอิมพีแดนซ์ที่เคอร์เซอร์ของสายต่าง ๆ ของระบบไฟฟ้ากำลัง ก่อนการคำนวณ เพียงผู้ใช้กด ENTER เท่านั้น จะปรากฏภาพดังรูปที่ 10

DATA FOR ZERO SEQUENCE

Number of Lines = 8			Number of Buses = 6			
LINE	BUS	BUS	ZERO		MUTUAL	LINE
2	0	2	0.0000	+0.0320j		
7	0	4	0.0000	+0.2660j		
9	0	6	0.0000	+0.6000j		
5	2	3	3.7800	+5.2600j		
6	2	5	2.8200	+3.8400j		
8	4	6	0.9000	+2.0600j		
3	4	1	0.8000	+1.8500j	0.5000	+0.9500j
4	1	6	0.9840	+2.0840j		8

รูปที่ 10

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

5.4 Z-BUS for Zero Seq. เป็นโมดย่อยที่แสดงค่าบัสสมมติแชนแนลเมตริกซ์ซีควนซ์ศูนย์ของระบบไฟฟ้ากำลัง เพียงผู้ใช้กด ENTER เท่านั้น จะปรากฏภาพดังรูปที่ 11

ZBUS FOR ZERO SEQUENCE			
BUS	BUS	R	X
1	1	0.36392	1.11336
1	2	0.0	0.0
1	3	0.0	0.0
1	4	-0.01120	0.11474
1	5	0.0	0.0
1	6	0.02526	0.34120
2	2	0.0	0.03200
2	3	0.0	0.03200
2	4	0.0	0.0
2	5	0.0	0.03200
2	6	0.0	0.0
3	3	3.78000	5.29200
3	4	0.0	0.0
3	5	0.0	0.03200
3	6	0.0	0.0
4	4	0.00756	0.24138
4	5	0.0	0.0
4	6	-0.01706	0.05554
5	5	2.82000	3.87200
5	6	0.0	0.0
6	6	0.03849	0.47472

Time used = 2.0 Seconds
Memory used = 40 Units

ศูนย์วิทยุ โทรทัศน์
จุฬาลงกรณ์มหาวิทยาลัย

5.5 Fault Solution เป็นโมดย่อยที่แสดงผลลัพธ์การวิเคราะห์การลัดวงจรเนื่องผู้ใช้กด ENTER เท่านั้น จะปรากฏดังรูปที่ 12-17 โดยที่จอภาพจะปรากฏผลลัพธ์ที่ละส่วนให้กดปุ่มใด ๆ ก็ได้ 1 ครั้ง จะปรากฏผลลัพธ์ในส่วนต่อไป หลังจากนั้นจะกลับสู่ Menu

SINGLE LINE TO GROUND FAULT AT BUS 1 REFERENCE PHASE : A

SEQUENCE QUANTITIES

BUS CURRENT

Bus	I0	ANG0	I1	ANG1	I2	ANG2
1	0.6263	-75.16	0.6263	-75.16	0.6263	-75.16

BUS VOLTAGE

Bus	V0	ANG0	V1	ANG1	V2	ANG2
1	0.7336	176.74	0.8665	-1.38	0.1354	-171.14
2	0.0000	0.00	0.9709	-0.69	0.0314	-158.18
3	0.0000	0.00	0.8983	-0.63	0.1023	-174.44
4	0.0722	-159.58	0.8920	-0.90	0.1091	-172.60
5	0.0000	0.00	0.9190	-0.48	0.0814	-174.54
6	0.2143	-169.39	0.8980	-0.97	0.1032	-171.54

LINE CURRENT

Bus to bus	I0	ANG0	I1	ANG1	I2	ANG2
0 1	0.0000	0.00	0.5623	-76.38	0.5623	-76.38
0 2	0.0000	0.00	0.0653	-64.60	0.0653	-64.60
4 1	0.3973	-76.08	0.0350	-62.95	0.0350	-62.95
6 1	0.2291	-73.56	0.0302	-66.52	0.0302	-66.52
2 3	0.0000	0.00	0.0285	-56.84	0.0285	-56.84
2 5	0.0000	0.00	0.0372	-70.54	0.0372	-70.54
3 4	0.0000	0.00	0.0285	-56.84	0.0285	-56.84
6 4	0.1313	-89.60	0.0074	-87.28	0.0074	-87.28
5 6	0.0000	0.00	0.0372	-70.54	0.0372	-70.54

จุฬาลงกรณ์มหาวิทยาลัย

SINGLE LINE TO GROUND FAULT AT BUS 2 REFERENCE PHASE : A

SEQUENCE QUANTITIES

BUS CURRENT

Bus	I0	ANGO	I1	ANG1	I2	ANG2
2	1.2519	-83.64	1.2519	-83.64	1.2519	-83.64

BUS VOLTAGE

Bus	V0	ANGO	V1	ANG1	V2	ANG2
1	0.0000	0.00	0.9391	-0.88	0.0627	-166.66
2	0.0401	-173.64	0.5199	0.24	0.4801	179.74
3	0.0401	-173.64	0.8078	-2.95	0.1977	-167.85
4	0.0000	0.00	0.8344	-1.95	0.1685	-170.29
5	0.0401	-173.64	0.7245	-3.13	0.2794	-171.85
6	0.0000	0.00	0.8108	-1.47	0.1906	-173.73

LINE CURRENT

Bus to bus	I0	ANGO	I1	ANG1	I2	ANG2
0 1	0.0000	0.00	0.2605	-71.90	0.2605	-71.90
0 2	1.2519	-83.64	0.9983	-86.69	0.9983	-86.69
1 4	0.0000	0.00	0.1399	-70.24	0.1399	-70.24
1 6	0.0000	0.00	0.1207	-73.81	0.1207	-73.81
3 2	0.0000	0.00	0.1138	-64.14	0.1138	-64.14
5 2	0.0000	0.00	0.1486	-77.83	0.1486	-77.83
3 4	0.0000	0.00	0.1138	115.86	0.1138	115.86
6 4	0.0000	0.00	0.0294	85.43	0.0294	85.43
5 6	0.0000	0.00	0.1486	102.17	0.1486	102.17

ศูนย์วิทยพัชกร
รูปที่ 13
จุฬาลงกรณ์มหาวิทยาลัย

SINGLE LINE TO GROUND FAULT AT BUS 3 REFERENCE PHASE : A

SEQUENCE QUANTITIES

BUS CURRENT

Bus	I0	ANGO	I1	ANG1	I2	ANG2
3	0.1263	-58.77	0.1263	-58.77	0.1263	-58.77

BUS VOLTAGE

Bus	V0	ANGO	V1	ANG1	V2	ANG2
1	0.0000	0.00	0.9809	-0.45	0.0206	-158.06
2	0.0040	-148.77	0.9841	-0.70	0.0199	-142.98
3	0.8214	175.69	0.9101	-1.94	0.0956	-161.17
4	0.0000	0.00	0.9332	-1.20	0.0698	-163.80
5	0.0040	-148.77	0.9668	-0.65	0.0350	-161.68
6	0.0000	0.00	0.9599	-0.81	0.0424	-161.31

LINE CURRENT

Bus to bus	I0	ANGO	I1	ANG1	I2	ANG2
0 1	0.0000	0.00	0.0856	-63.29	0.0856	-63.29
0 2	0.1263	-58.77	0.0415	-49.40	0.0415	-49.40
1 4	0.0000	0.00	0.0651	-64.00	0.0651	-64.00
1 6	0.0000	0.00	0.0205	-61.03	0.0205	-61.03
2 3	0.1263	-58.77	0.0302	-41.26	0.0302	-41.26
2 5	0.0000	0.00	0.0124	-69.57	0.0124	-69.57
4 3	0.0000	0.00	0.0980	-64.08	0.0980	-64.08
6 4	0.0000	0.00	0.0328	-64.24	0.0328	-64.24
5 6	0.0000	0.00	0.0124	-69.57	0.0124	-69.57

รูปที่ 14
ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



SINGLE LINE TO GROUND FAULT AT BUS 4 REFERENCE PHASE : A

SEQUENCE QUANTITIES

BUS CURRENT

Bus	I0	ANGO	I1	ANG1	I2	ANG2
4	0.7034	-78.93	0.7034	-78.93	0.7034	-78.93

BUS VOLTAGE

Bus	V0	ANGO	V1	ANG1	V2	ANG2
1	0.0811	-163.36	0.8778	-0.51	0.1225	-176.37
2	0.0000	0.00	0.9086	-1.49	0.0947	-165.58
3	0.0000	0.00	0.6128	2.51	0.3887	176.04
4	0.1699	-170.73	0.5840	1.34	0.4164	178.12
5	0.0000	0.00	0.7992	0.53	0.2009	177.91
6	0.0409	-151.85	0.7510	0.05	0.2490	179.86

LINE CURRENT

Bus to bus	I0	ANGO	I1	ANG1	I2	ANG2
0 1	0.0000	0.00	0.5086	-81.61	0.5086	-81.61
0 2	0.0000	0.00	0.1968	-72.00	0.1968	-72.00
1 4	0.0182	-59.31	0.3893	-81.97	0.3893	-81.97
1 6	0.0182	120.69	0.1193	-80.41	0.1193	-80.41
2 3	0.0000	0.00	0.1178	-65.12	0.1178	-65.12
2 5	0.0000	0.00	0.0811	-82.04	0.0811	-82.04
3 4	0.0000	0.00	0.1178	-65.12	0.1178	-65.12
6 4	0.0500	-62.78	0.2004	-81.07	0.2004	-81.07
5 6	0.0000	0.00	0.0811	-82.04	0.0811	-82.04
0 4	0.6386	-80.73	0.0000	0.00	0.0000	0.00

ศูนย์วิทยทรัพยากร
รูปที่ 15
จุฬาลงกรณ์มหาวิทยาลัย

SINGLE LINE TO GROUND FAULT AT BUS 5 REFERENCE PHASE : A

SEQUENCE QUANTITIES

BUS CURRENT

Bus	I0	ANGO	I1	ANG1	I2	ANG2
5	0.1581	-60.12	0.1581	-60.12	0.1581	-60.12

BUS VOLTAGE

Bus	V0	ANGO	V1	ANG1	V2	ANG2
1	0.0000	0.00	0.9808	-0.42	0.0205	-159.50
2	0.0051	-150.12	0.9702	-1.09	0.0353	-148.33
3	0.0051	-150.12	0.9582	-0.76	0.0438	-163.03
4	0.0000	0.00	0.9568	-0.78	0.0452	-163.28
5	0.7572	173.81	0.8774	-2.67	0.1302	-161.73
6	0.0000	0.00	0.9260	-1.14	0.0765	-166.02

LINE CURRENT

Bus to bus	I0	ANGO	I1	ANG1	I2	ANG2
0 1	0.0000	0.00	0.0853	-64.73	0.0853	-64.73
0 2	0.1581	-60.12	0.0734	-54.75	0.0734	-54.75
1 4	0.0000	0.00	0.0326	-64.23	0.0326	-64.23
1 6	0.0000	0.00	0.0527	-65.04	0.0527	-65.04
2 3	0.0000	0.00	0.0052	-81.27	0.0052	-81.27
2 5	0.1581	-60.12	0.0688	-52.83	0.0688	-52.83
3 4	0.0000	0.00	0.0052	-81.27	0.0052	-81.27
6 4	0.0000	0.00	0.0376	113.46	0.0376	113.46
6 5	0.0000	0.00	0.0903	-65.67	0.0903	-65.67

ศูนย์วิทยุทรัพยากร
รูปที่ 16
จุฬาลงกรณ์มหาวิทยาลัย

SINGLE LINE TO GROUND FAULT AT BUS 6 REFERENCE PHASE : A

SEQUENCE QUANTITIES

BUS CURRENT

Bus	I0	ANG0	I1	ANG1	I2	ANG2
6	0.5803	-80.00	0.5803	-80.00	0.5803	-80.00

BUS VOLTAGE

Bus	V0	ANG0	V1	ANG1	V2	ANG2
1	0.1985	-174.24	0.9046	-0.38	0.0956	-176.38
2	0.0000	0.00	0.9131	-0.95	0.0883	-170.09
3	0.0000	0.00	0.8053	0.62	0.1949	177.45
4	0.0337	-152.93	0.7946	0.31	0.2054	178.79
5	0.0000	0.00	0.7213	2.29	0.2808	174.10
6	0.2764	-174.64	0.6377	1.16	0.3626	177.96

LINE CURRENT

Bus to bus	I0	ANG0	I1	ANG1	I2	ANG2
0 1	0.0000	0.00	0.3971	-81.62	0.3971	-81.62
0 2	0.0000	0.00	0.1837	-76.51	0.1837	-76.51
1 4	0.0338	119.62	0.1459	-83.19	0.1459	-83.19
1 6	0.0338	-60.38	0.2514	-80.70	0.2514	-80.70
2 3	0.0000	0.00	0.0433	-67.94	0.0433	-67.94
2 5	0.0000	0.00	0.1410	-79.13	0.1410	-79.13
3 4	0.0000	0.00	0.0433	-67.94	0.0433	-67.94
4 6	0.0930	-63.85	0.1879	-79.72	0.1879	-79.72
5 6	0.0000	0.00	0.1410	-79.13	0.1410	-79.13
0 6	0.4606	-84.64	0.0000	0.00	0.0000	0.00

รูปที่ 17
ศูนย์วิทยพัชการ
จุฬาลงกรณ์มหาวิทยาลัย

ในการนี้ผู้ใช้เลือกใช้โปรแกรม FP โมดการทำงานดังกล่าวจะปรากฏภาพดังรูปที่ 18

Cell: 1A

File: S2

MENU

Worksheet		File		Calculate		Display		Print		Exit	
Line	Bus		Equip.	Type	Self Imped			Zoning		Mutual Impedance	
	P	Q			Positive			Z1, Z2, Y4	Z-BUS	Zm	Line
1	0	1	GEN	Y	0.0200	0.2400	0				
2	0	2	GEN	YG	0.0300	0.4800	0.0000	0.0320			
3	1	4	LINE		0.1600	0.7400	0.8000	1.8500			
4	1	6	LINE		0.2460	1.0360	0.9840	2.0840			
5	2	3	LINE		1.4460	2.1000	3.7800	5.2600			
6	2	5	LINE		0.5640	1.2800	2.8200	3.8400			
7	3	4	XMER	D-YG	0.0000	0.2660	0.0000	0.2660			
8	6	4	LINE		0.1940	0.8140	0.9000	2.0600	0.5000	0.9500	3
9	5	6	XMER	D-YG	0.0000	0.6000	0.0000	0.6000			

รูปที่ 18

5.6 Zoning เป็นโมดย่อยที่แสดงการแบ่งระบบย่อยของระบบไฟฟ้ากำลังก่อนการคำนวณ
เพียงผู้ใช้กด ENTER เท่านั้น จะปรากฏภาพดังรูปที่ 19

LIST OF ZONES

BUSES IN ZONE A : 1 4 6

BUSES IN ZONE B : 2 3 5

LIST OF TIE LINE : 3 - 4 5 - 6

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

รูปที่ 19

5.7 Z1, Z2, Y4 เป็นโมดย่อยที่แสดงบัลลิมินแดนซ์เมตริกซ์ของระบบย่อย และเมตริกซ์
เชื่อมโยงระบบย่อย เพียงผู้ใช้กด ENTER เท่านั้น จะปรากฏภาพดังรูปที่ 20

Z1 MATRIX FOR ZONE A

BUS	BUS	R	X
1	1	0.02000	0.24000
1	4	0.02000	0.24000
1	6	0.02000	0.24000
4	4	0.13754	0.76862
4	6	0.08571	0.53603
6	6	0.16520	0.86161

Z1 MATRIX FOR ZONE B

BUS	BUS	R	X
2	2	0.03000	0.48000
2	3	0.03000	0.48000
2	5	0.03000	0.48000
3	3	1.47600	2.58000
3	5	0.03000	0.48000
5	5	0.59400	1.76000

Z2 MATRIX FOR ZONE A

BUS	TIE LINE	R	X
1	3 - 4	0.02000	0.24000j
1	5 - 6	0.02000	0.24000j
4	3 - 4	0.13754	0.76862j
4	5 - 6	0.08571	0.53603j
6	3 - 4	0.08571	0.53603j
6	5 - 6	0.16520	0.86161j

 Z2 MATRIX FOR ZONE B

BUS	TIE LINE	R	X
2	3 - 4	-0.03000	-0.48000j
2	5 - 6	-0.03000	-0.48000j
3	3 - 4	-1.47600	-2.58000j
3	5 - 6	-0.03000	-0.48000j
5	3 - 4	-0.03000	-0.48000j
5	5 - 6	-0.59400	-1.76000j

 Y4 MATRIX

TIE LINE	TIE LINE	R	X
3 - 4	3 - 4	0.11972	-0.24454j
3 - 4	5 - 6	-0.04563	0.07067j
5 - 6	5 - 6	0.08587	-0.31409j

Time used = 1.2 Sec. Memory used = 40 Units

ศูนย์วิทยทรัพยากร
 รพที่ 20 (ต่อ)
 จุฬาลงกรณ์มหาวิทยาลัย

5.8 Z-BUS เป็นโมดย่อยที่แสดงบัสอิมพีแดนซ์เมตริกซ์ของระบบไฟฟ้ากำลัง ที่เกิดจากระบบย่อย เพียงผู้ใช้กด ENTER เท่านั้น จะปรากฏภาพดังรูปที่ 21

ZBUS FOR POSITIVE SEQUENCE			
BUS	BUS	R	X
1	1	0.02253	0.21503
1	2	-0.00609	0.04974
1	3	0.02635	0.16117
1	4	0.02254	0.17266
1	5	0.02118	0.12825
1	6	0.01831	0.16379
2	2	0.04422	0.38094
2	3	-0.01594	0.15713
2	4	-0.00786	0.13434
2	5	-0.00698	0.22306
2	6	0.00023	0.15223
3	3	0.16244	0.73912
3	4	0.14333	0.53368
3	5	0.06192	0.27007
3	6	0.07295	0.32786
4	4	0.13269	0.57694
4	5	0.06506	0.27818
4	6	0.06881	0.34726
5	5	0.16569	0.80649
5	6	0.13256	0.46538
6	6	0.13034	0.61119

ศูนย์วิทยทรัพยากร จุฬาลงกรณ์มหาวิทยาลัย

รูปที่ 21

5.9 Fault Solution เป็นโมดย่อยที่แสดงผลการวิเคราะห์การลัดวงจรโดยวิธีแยกเป็นส่วนย่อย เพียงผู้ใช้กด ENTER เท่านั้น จะปรากฏภาพดังรูปที่ 22-27

BALANCE THREE PHASE FAULT AT BUS 1 REFERENCE PHASE : A

SEQUENCE QUANTITIES

BUS CURRENT

Bus	I0	ANG0	I1	ANG1	I2	ANG2
1	0.0000	0.00	4.6251	-84.02	0.0000	0.00

BUS VOLTAGE

Bus	V0	ANG0	V1	ANG1	V2	ANG2
1	0.0000	0.00	0.0000	0.00	0.0000	0.00
4	0.0000	0.00	0.1960	5.99	0.0000	0.00
6	0.0000	0.00	0.2378	1.27	0.0000	0.00

LINE CURRENT

Bus to bus	I0	ANG0	I1	ANG1	I2	ANG2
0 1	0.0000	0.00	4.1523	-85.24	0.0000	0.00
4 1	0.0000	0.00	0.2588	-71.80	0.0000	0.00
6 1	0.0000	0.00	0.2233	-75.38	0.0000	0.00

รูปที่ 22

BALANCE THREE PHASE FAULT AT BUS 2 REFERENCE PHASE : A

SEQUENCE QUANTITIES

BUS CURRENT

Bus	I0	ANG0	I1	ANG1	I2	ANG2
2	0.0000	0.00	2.6076	-83.38	0.0000	0.00

BUS VOLTAGE

Bus	V0	ANG0	V1	ANG1	V2	ANG2
2	0.0000	0.00	0.0000	0.00	0.0000	0.00
3	0.0000	0.00	0.6043	-8.42	0.0000	0.00
5	0.0000	0.00	0.4328	-11.35	0.0000	0.00

LINE CURRENT

Bus to bus	I0	ANG0	I1	ANG1	I2	ANG2
0 2	0.0000	0.00	2.0793	-86.42	0.0000	0.00
3 2	0.0000	0.00	0.2370	-63.87	0.0000	0.00
5 2	0.0000	0.00	0.3094	-77.57	0.0000	0.00

รูปที่ 23

BALANCE THREE PHASE FAULT AT BUS 3 REFERENCE PHASE : A

SEQUENCE QUANTITIES

BUS CURRENT		I0	ANGO	I1	ANG1	I2	ANG2
Bus	3	0.0000	0.00	1.3214	-77.61	0.0000	0.00
BUS VOLTAGE		V0	ANGO	V1	ANG1	V2	ANG2
Bus	2	0.0000	0.00	0.8044	-4.64	0.0000	0.00
	3	0.0000	0.00	0.0000	0.00	0.0000	0.00
	4	0.0000	0.00	0.2726	7.08	0.0000	0.00
LINE CURRENT		I0	ANGO	I1	ANG1	I2	ANG2
Bus to bus	2 3	0.0000	0.00	0.3155	-60.09	0.0000	0.00
	4 3	0.0000	0.00	1.0250	-82.92	0.0000	0.00

รูปที่ 24

BALANCE THREE PHASE FAULT AT BUS 4 REFERENCE PHASE : A

SEQUENCE QUANTITIES

BUS CURRENT		I0	ANGO	I1	ANG1	I2	ANG2
Bus	4	0.0000	0.00	1.6892	-77.05	0.0000	0.00
BUS VOLTAGE		V0	ANGO	V1	ANG1	V2	ANG2
Bus	1	0.0000	0.00	0.7078	-2.29	0.0000	0.00
	3	0.0000	0.00	0.0752	26.77	0.0000	0.00
	4	0.0000	0.00	0.0000	0.00	0.0000	0.00
	6	0.0000	0.00	0.4027	-2.59	0.0000	0.00
LINE CURRENT		I0	ANGO	I1	ANG1	I2	ANG2
Bus to bus	1 4	0.0000	0.00	0.9349	-80.09	0.0000	0.00
	3 4	0.0000	0.00	0.2829	-63.23	0.0000	0.00
	6 4	0.0000	0.00	0.4812	-79.19	0.0000	0.00

รูปที่ 25

BALANCE THREE PHASE FAULT AT BUS 5 REFERENCE PHASE : A

SEQUENCE QUANTITIES

BUS CURRENT		I0	ANG0	I1	ANG1	I2	ANG2
Bus							
5		0.0000	0.00	1.2146	-78.39	0.0000	0.00

BUS VOLTAGE		V0	ANG0	V1	ANG1	V2	ANG2
Bus							
2		0.0000	0.00	0.7390	-4.88	0.0000	0.00
5		0.0000	0.00	0.0000	0.00	0.0000	0.00
6		0.0000	0.00	0.4163	6.06	0.0000	0.00

LINE CURRENT		I0	ANG0	I1	ANG1	I2	ANG2
Bus to bus							
2	5	0.0000	0.00	0.5283	-71.10	0.0000	0.00
6	5	0.0000	0.00	0.6938	-83.94	0.0000	0.00

รูปที่ 26

BALANCE THREE PHASE FAULT AT BUS 6 REFERENCE PHASE : A

SEQUENCE QUANTITIES

BUS CURRENT		I0	ANG0	I1	ANG1	I2	ANG2
Bus							
6		0.0000	0.00	1.6002	-77.96	0.0000	0.00

BUS VOLTAGE		V0	ANG0	V1	ANG1	V2	ANG2
Bus							
1		0.0000	0.00	0.7380	-2.02	0.0000	0.00
4		0.0000	0.00	0.4337	-1.09	0.0000	0.00
5		0.0000	0.00	0.2334	12.91	0.0000	0.00
6		0.0000	0.00	0.0000	0.00	0.0000	0.00

LINE CURRENT		I0	ANG0	I1	ANG1	I2	ANG2
Bus to bus							
1	6	0.0000	0.00	0.6931	-78.66	0.0000	0.00
4	6	0.0000	0.00	0.5182	-77.68	0.0000	0.00
5	6	0.0000	0.00	0.3889	-77.09	0.0000	0.00

รูปที่ 27

6. Print

Cell: 1A

File: S2

MENU

Worksheet		File		Calculate		Display		Print		Exit	
Line	Bus		Equip.	Type	Self Impedance				Input Data		
	P	Q			Positive		Zero		Data for Positive Seq.	Z-BUS for Positive Seq.	Data for Zero Seq.
1	0	1	GEN	Y	0.0200	0.2400	0.0000	0.			
2	0	2	GEN	YG	0.0300	0.4800	0.0000	0.			
3	1	4	LINE		0.1600	0.7400	0.8000	1.			
4	1	6	LINE		0.2460	1.0360	0.9840	2.0840			
5	2	3	LINE		1.4460	2.1000	3.7800	5.2600			
6	2	5	LINE		0.5640	1.2800	2.8200	3.8400			
7	3	4	XMER	D-YG	0.0000	0.2660	0.0000	0.2660			
8	6	4	LINE		0.1940	0.8140	0.9000	2.0600	0.5000	0.9500	3
9	5	6	XMER	D-YG	0.0000	0.6000	0.0000	0.6000			
10											

รูปที่ 28

Cell: 1A

File: S2

MENU

Worksheet		File		Calculate		Display		Print		Exit	
Line	Bus		Equip.	Type	Self Impedance				Input Data		
	P	Q			Positive		Zero		Zoning	Z1, Z2, Y4	Z-BUS
1	0	1	GEN	Y	0.0200	0.2400	0.0000	0.0320			
2	0	2	GEN	YG	0.0300	0.4800	0.0000	0.0320			
3	1	4	LINE		0.1600	0.7400	0.8000	1.8500			
4	1	6	LINE		0.2460	1.0360	0.9840	2.0840			
5	2	3	LINE		1.4460	2.1000	3.7800	5.2600			
6	2	5	LINE		0.5640	1.2800	2.8200	3.8400			
7	3	4	XMER	D-YG	0.0000	0.2660	0.0000	0.2660			
8	6	4	LINE		0.1940	0.8140	0.9000	2.0600	0.5000	0.9500	3
9	5	6	XMER	D-YG	0.0000	0.6000	0.0000	0.6000			
10											

รูปที่ 29

ดังแสดงในรูปที่ 28 และ รูปที่ 29 เป็นโมดการทำงานที่ใช้สำหรับพิมพ์ผลลัพธ์ การวิเคราะห์ต่าง ๆ ของทั้งสองโปรแกรม โดยจะแสดงภาพเช่นเดียวกับที่แสดงบนจอภาพ (Display) เพียงแต่ผู้ใช้กด ENTER หลังจากเลือกโมดย่อยแล้ว

7. Exit

Cell: 1A

File: S2

MENU

Line	Worksheet		Equip.	Type	Calculate				Mutual Zm	Exit	
	P	Q			Self Impedance		Zm	No		ce	
					Positive	Zero		Yes		ine	
1	0	1	GEN	Y	0.0200	0.2400	0.0000	0.0320			
2	0	2	GEN	YG	0.0300	0.4800	0.0000	0.0320			
3	1	4	LINE		0.1600	0.7400	0.8000	1.8500			
4	1	6	LINE		0.2460	1.0360	0.9840	2.0840			
5	2	3	LINE		1.4460	2.1000	3.7800	5.2600			

รูปที่ 30

ดังแสดงในรูปที่ 30 เป็นโมดการทำงานเพื่อให้ผู้ใช้ติการวิเคราะห์การลัดวงจร โดยเลือกตัวชี้เซลไปที่ YES, NO แล้วกด ENTER เท่านั้น ถ้าเลือก YES โปรแกรมจะกลับไป DOS ถ้าเลือก NO โปรแกรมจะกลับไป Frame Work ดังรูปที่ 1

8. การป้อนและแก้ไขข้อมูล

Cell: 7D

File: S2

JOBS

OldData: D-YG

Date: 06-14-62

Require: YG-YG, YG-Y, Y-YG, YG-D, D-YG, Y-D, D-Y or D-D

Time: 06:00:00

Line	Bus		Equip.	Type	Self Impedance				Mutual Impedance	
	P	Q			Positive		Zero	Zm	Line	
1	0	1	GEN	Y	0.0200	0.2400	0.0000	0.0320		
2	0	2	GEN	YG	0.0300	0.4800	0.0000	0.0320		
3	1	4	LINE		0.1600	0.7400	0.8000	1.8500		
4	1	6	LINE		0.2460	1.0360	0.9840	2.0840		
5	2	3	LINE		1.4460	2.1000	3.7800	5.2600		
6	2	5	LINE		0.5640	1.2800	2.8200	3.8400		
7	3	4	XMER	D-YG	0.0000	0.2660	0.0000	0.2660		
8	6	4	LINE		0.1940	0.8140	0.9000	2.0600	0.5000	0.9500 3
9	5	6	XMER	D-YG	0.0000	0.6000	0.0000	0.6000		
10										
11										
12										
13										
14										
15										

F2-Save

F6-Print

F10,/-MEMU

รูปที่ 31

ดังแสดงในรูปที่ 31 การที่จะป้อนหรือแก้ไขค่าต่าง ๆ ลงในแต่ละเซลล์ได้ จะต้องสามารถเลื่อนเซลล์ไปยังตำแหน่งใด ๆ ได้ โดยการกดปุ่มที่มีสัญลักษณ์ " " = ขึ้น, " " = ลง, " " = ไปซ้าย, " " = ไปขวา บรรทัดแรกด้านซ้ายของรูปที่ 31 จะแสดงตำแหน่งของเซลล์

เมื่อเลื่อนตัวชี้เซลล์ไปยังตำแหน่งเซลล์ต่าง ๆ บรรทัดที่สองด้านซ้ายของรูปที่ 31 จะแสดงข้อมูลเดิมของเซลล์นั้น บรรทัดที่สามด้านซ้ายของรูปที่ 30 จะแสดงชนิดและขนาดของข้อมูลที่จะป้อน ดังนี้

- ตำแหน่ง NA (N = 1,100) กำหนดให้ป้อนตัวเลขตั้งแต่ 0 ถึง 49
- ตำแหน่ง NB กำหนดให้ป้อนตัวเลขตั้งแต่ 1 ถึง 49
- ตำแหน่ง NC กำหนดให้ป้อนตัวหนังสือ GEN, XMER, LINE
- ตำแหน่ง ND กำหนดให้ป้อนตัวหนังสือ โดยมีข้อกำหนดดังนี้
 - ถ้าตำแหน่ง NC เป็น GEN จะกำหนดให้ป้อนตัวหนังสือ Y, YG, D
 - ถ้าตำแหน่ง NC เป็น XMER จะกำหนดให้ป้อนตัวหนังสือ YG-YG, YG-Y, Y-YG, YG-D, D-YG, Y-D, D-Y, D-D
 - ถ้าตำแหน่ง NC เป็น LINE จะไม่ต้องป้อนค่าใด ๆ
- ตำแหน่ง NE กำหนดให้ป้อนตัวเลข 0 ถึง -9.9999
- ตำแหน่ง NF กำหนดให้ป้อนตัวเลข - 9.9999 ถึง 9.9999
- ตำแหน่ง NG กำหนดให้ป้อนตัวเลข 0 ถึง 9.9999
- ตำแหน่ง NH กำหนดให้ป้อนตัวเลข -9.9999 ถึง 9.9999
- ตำแหน่ง NI กำหนดให้ป้อนตัวเลข 0 ถึง -9.9999
- ตำแหน่ง NJ กำหนดให้ป้อนตัวเลข -9.9999 ถึง 9.999
- ตำแหน่ง NK กำหนดให้ป้อนตัวเลข 1 ถึง 99

ในกรณีที่ป้อนค่าต่าง ๆ ผิดจากที่กำหนดไว้ บรรทัดสุดท้ายของรูปที่ 31 จะปรากฏข้อความ เช่น

" Error " Require only Number, Minus sign or Point

ให้กดปุ่มใด ๆ ก็ได้โปรแกรมจะกลับไปขึ้นขั้นตอนก่อนป้อนข้อมูลดังกล่าว ผู้ใช้จะสามารถป้อนค่าใหม่ได้ หรือหากพิมพ์ข้อมูลผิดไปตัวหนึ่ง ก็สามารถลบออกได้ทันที โดยใช้ Back Space

เมื่อผู้ใช้ป้อนหรือแก้ไขข้อมูลทุกครั้งเพียงเลื่อนตัวชี้เซลล์ไป ข้อมูลดังกล่าวก็จะถูกบันทึกไว้ หรือผู้ใช้อาจกด ENTER ก็ได้

การป้อนข้อมูลของลายหน้าที่ 1 โปรแกรมจะกำหนดให้ลายหน้าจะต้องเป็นเครื่องกำเนิดไฟฟ้า ผู้ใช้จะต้องป้อนข้อมูลที่ตำแหน่ง 1A ด้วยค่า 0 (ศูนย์) เสมอ มิฉะนั้นเครื่องจะฟ้องข้อผิดพลาด

11. โมดการทำงาน

ดังกล่าวถึงในหัวข้อที่ 1 ส่วนที่ 3 ของ Frame Work เป็นโมดการทำงานของโปรแกรมในส่วนที่ไม่เกี่ยวกับการคำนวณ อธิบายได้ดังนี้

- F2-Save หมายถึงการเก็บข้อมูลที่ปรากฏบนจอภาพเข้าไปในไฟล์ชื่อเดิม
- F5-Print หมายถึงการพิมพ์ข้อมูลที่ปรากฏบนจอภาพ
- F10,/ -Menu หมายถึงการเข้าสู่โมดการทำงานของโปรแกรม

ในกรณีที่เครื่องกำลังทำงานอยู่ในโมดการทำงาน ถ้าผู้ใช้ต้องการออกจากโมดการทำงานเข้าสู่ Frame Work โดยเพื่อตรวจสอบหรือแก้ไขข้อมูล ทำได้โดยกดปุ่ม "Esc"



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



ภาคผนวก ข.

โปรแกรมการวิเคราะห์การคลังวงจร โดยวิบีเอสอิมพินดอร์เมตวิกซ์

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```

/* ----- */
/*                                     */
/*      File : FA.A                    */
/*                                     */
/*      Individual file for : FA.EXE   */
/*                                     */
/*      TITLE routine for : FA1.EXE   */
/*                                     */
/* ----- */

#include <stdio.h>
#include <conio.h>
#include <process.h>

char Title[ ] =
"
"
"      <----->
"      DEPARTMENT OF ELECTRICAL ENGINEERING
"      FACULTY OF ENGINEERING
"      CHULALONGKORN UNIVERSITY
"      {-----}
"      COMPUTER AIDED
"      FAULT ANALYSIS
"      {-----}
"      BY
"      THANA CHATAVARAHA M.ENG.(E.E.)
"      [-----]
"
"Loading Program . . . Wait a minute."
;

/* ----- */

main()
{
    int i;
    struct text_info monotor;
    gettextinfo(&monotor);
    textmode(monotor.currmode);
    for (i=0; Title[i]>0; ++i)
        switch (Title[i]) {
            case '<': Title[i] = 201; break;

```



```
case '>': Title[i] = 187; break;
case '{': Title[i] = 199; break;
case ')': Title[i] = 182; break;
case '[': Title[i] = 200; break;
case ']': Title[i] = 188; break;
case '=': Title[i] = 205; break;
case '-': Title[i] = 196; break;
case '!': Title[i] = 186; break;
}
textattr(GREEN+(BLACK<<4)), cputs(Title);
if (monotor.currmode == 7) execl( "FA1.EXE", "", "mono", NULL);
else execl( "FA1.EXE", "", "color", NULL);
}
/*-----*/
```



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

MAIN.OBJ
MAUX.OBJ
MOVE.OBJ
MPUT.OBJ
MENU.OBJ
NO_PLEX.OBJ
N1_ORDER.OBJ
N2_MUTU.OBJ
N3_ZBUS.OBJ
N4_FAULT.OBJ



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```

/* ----- */
/*                                     */
/*   File: HAT.B   -->   Head Table Definition   */
/*                                     */
/* ----- */

```

```

#define HAT_1      "Cell: 1A"                File:                                READY \n"
#define HAT_2      "OldData: Space"          Date: 06-14-62\n"
#define HAT_3      "Require: Pressing Key"    Time: 06:00:00\n"
#define HAT_4      "-----\n"
#define HAT_5      "   | Bus                Self Impedance          Mutual Impedance |\n"
#define HAT_6      "   |Line| ----- Equip. Type ----- |\n"
#define HAT_7      "   |   | P Q                Positive          Zero          Zm          Line |\n"
#define HAT_8      "   |-----|\n"
#define HAT_8      "   |-----|\n"
#define BLANK      "   | |\n"
#define BLANK      "   |-----|\n"
#define BOTTOM     "-----\n"

```

```

#define REQ_1      "NUMBER; 0 to 49"
#define REQ_2      "NUMBER; 1 to 50"
#define REQ_3      "GEN, XMER or LINE"
#define REQ_4a     "Y or YG"
#define REQ_4b     "YG-YG, YG-Y, Y-YG, YG-D, D-YG, Y-D, D-Y or D-D"
#define REQ_4c     " ( BLANK / SPACE ) "
#define REQ_5      "VALUE; 0 to 9.9999"
#define REQ_6      "VALUE; -9.9999 to 9.9999"
#define REQ_7      "VALUE; 0 to 9.9999"
#define REQ_8      "VALUE; -9.9999 to 9.9999"
#define REQ_9      "VALUE; 0 to 9.9999"
#define REQ_10     "VALUE; -9.9999 to 9.9999"
#define REQ_11     "NUMBER; 1 to 99"

```

```

#define DAT_00     "                                DATA FOR ZERO SEQUENCE                                \n"
#define DAT_01     "-----\n"
#define DAT_02     "Number of Lines =%3d                                Number of Buses =%3d\n"
#define DAT_03     "-----\n"
#define DAT_04     "LINE   BUS   BUS   ZERO   MUTUAL   LINE\n"

```

```

#define DAT_10     "                                DATA FOR POSITIVE SEQUENCE                                \n"
#define DAT_11     "-----\n"
#define DAT_12     "Number of Lines =%3d                                Number of Buses =%3d\n"
#define DAT_13     "-----\n"
#define DAT_14     "LINE   BUS   BUS   POSITIVE\n"

```

```

#define FORM_1     "%4d   %3d   %3d   %8.4f %8.4fj\n"
#define FORM_2     "%4d   %3d   %3d   %8.4f %8.4fj   %8.4f %8.4fj   %3d \n"

```

```

#define ZB_00     "-----\n"
#define ZB_10     "                                ZBUS FOR ZERO SEQUENCE                                \n"
#define ZB_11     "                                ZBUS FOR POSITIVE SEQUENCE                                \n"
#define ZB_20     "-----\n"
#define ZB_30     "BUS   BUS   R   X   \n"

```

```

#define FORM_3  "x3d  x3d          0.0      0.0\n"
#define FORM_4  "x3d  x3d          0.0      x#9.5f\n"
#define FORM_5  "x3d  x3d          x#9.5f   0.0\n"
#define FORM_6  "x3d  x3d          x#9.5f   x#9.5f\n"

#define OUT_00  " BALANCE THREE PHASE FAULT AT BUS x3d  REFERENCE PHASE :  %c  \n"
#define OUT_01  " LINE TO LINE FAULT AT BUS x3d  REFERENCE PHASE :  %c  \n"
#define OUT_02  " SINGLE LINE TO GROUND FAULT AT BUS x3d  REFERENCE PHASE :  %c  \n"
#define OUT_03  " DOUBLE LINE TO GROUND FAULT AT BUS x3d  REFERENCE PHASE :  %c  \n"
#define OUT_    "-----\n"

#define OUT_1   "Bus          IO      ANGO      I1      ANG1      I2      ANG2\n"
#define OUT_1P  "Bus          IA      ANGA      IB      ANGB      IC      ANGC\n"

#define OUT_2   "Bus          VO      ANGO      V1      ANG1      V2      ANG2\n"
#define OUT_2P  "Bus          VA      ANGA      VB      ANGB      VC      ANGC\n"

#define OUT_3   "Bus to bus   IO      ANGO      I1      ANG1      I2      ANG2\n"
#define OUT_3P  "Bus to bus   IA      ANGA      IB      ANGB      IC      ANGC\n"

#define FORM_7  "x3d          x#8.4f x#8.2f   x#8.4f x#8.2f   x#8.4f x#8.2f\n"
#define FORM_8  "x3d  x3d     x#8.4f x#8.2f   x#8.4f x#8.2f   x#8.4f x#8.2f\n"

/*-----*/

```



คุรุณย์วิทยทรพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```

/* ----- */
/*                                     */
/*   File: KEY.B   -->   Keyboard Definition   */
/*                                     */
/* ----- */

#define BS      8      /* Back space */
#define TAB     9      /* Tab        */
#define ENTER  13     /* Enter/Return */
#define ESC    27     /* Esc (Escape) */

#define F1     315    /* Function key */
#define F2     316
#define F3     317
#define F4     318
#define F5     319
#define F6     320
#define F7     321
#define F8     322
#define F9     323
#define F10    324

#define HOME   327    /* Move key */
#define UP     328
#define PGUP   329
#define LEFT   331
#define RIGHT  333
#define END    335
#define DOWN   336
#define PGDN   337

#define INS    338    /* Insert key */
#define DEL    339    /* Delete key */
#define EXIT   350    /* Ctrl F1 Program Exit */

#define GREEN  2      /* normal Green color */
#define LIGHT  10     /* Light green color */
#define REVER  120    /* Reverse green color */
#define FLASH  130    /* Flashing green color */

/* ----- */

```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย


```

/* ----- */
/*
/*   File : MAX.B == Maximum Definition for FP1.EXE
/*
/* ----- */

#define MFLD    12      /* Max. Field for I/P */
#define SIZE    82      /* Size of each record */

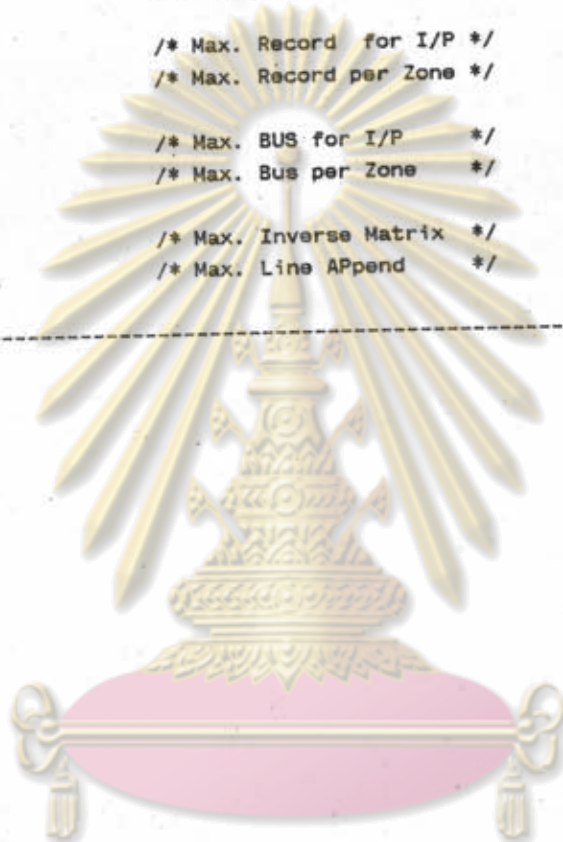
#define MREC    99      /* Max. Record for I/P */
#define MRZ     55      /* Max. Record per Zone */

#define MBUS    50      /* Max. BUS for I/P */
#define MBZ     30      /* Max. Bus per Zone */

#define MI      4       /* Max. Inverse Matrix */
#define MLAP    20      /* Max. Line Append */

/* ----- */

```



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```

/* ----- */
/*
/*   File : MAIN.C
/*
/*
/*   MAIN PROGRAM for : FA1.EXE
/*
/* ----- */

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <math.h>
#define COMPLEX struct complex

#include "HAT.B"
#include "KEY.B"
#include "MAX.B"

#define eif    else if

#define D(r,f) &data[(r)][fldaddr[(f)]]    /* addr.Data in rec,fld */
#define T(r,f) strset( temp, 0),strncpy( temp, D((r),(f)), fldlen[(f)] )

#define CINV   fldaddr[fld]                /* Column INVerse color */
#define CDAT   (CINV) + 1                  /* Column present Data */
#define LDAT   fldlen[fld]                 /* Length present Data */
#define LINV   (LDAT) + 2                  /* Length INVerse color */

static char   cell[4],
              temp[15],                    /* temporary string */
              data[MREC][SIZE];           /* data in table */
static int    i, j, NINP,                  /* Number of Input rec. */
              row = 9, rec = 0,
              top = 0, fld = 1,
              fldaddr[MFLD] = {2, 7,10,14,20,26,33,43,50,60,67,76},
              fldtype[MFLD] = {2, 2, 2, 1, 1, 3, 4, 3, 4, 3, 4, 2},
              fldint[MFLD] = {2, 2, 2, 4, 5, 1, 2, 1, 2, 1, 2, 2},
              fldmod[MFLD] = {0, 0, 0, 0, 0, 4, 4, 4, 4, 4, 4, 0},
              fldlen[MFLD] = {2, 2, 2, 4, 5, 6, 7, 6, 7, 6, 7, 2};

static FILE   *fx;

int           KEY,                          /* Keyboard */
              ERR,                          /* error status */
              NBUS,                          /* Number of Buses */
              NLINE,                         /* Number of used Lines */
              ReadData,
              SaveData,
              FormZbus,                      /* Form new Zbus or not */
              eq[MREC],                     /* Equipment */
              ty[MREC],                      /* type */
              lm[MREC],                     /* line mutual */
              bp[2][MREC],                  /* Low bus No. (P) */
              bq[2][MREC],                  /* High bus No. (Q) */
              COMPLEX zp[2][MREC],          /* primitive impedance */

```

```

                zmutu[MREC];                /* mutual impedance */
        char      filename[15];            /* data file name */

/* ----- */

main(int ac, char *av[])
{
    ReadData = ac;
    SET_VIDEO( av[1] );                /* MAUX */
    FormZbus = 1;
    SET_MENU();                        /* MENU */
    SET_FAR();                          /* ZBUS */
    CLEAR_DATA();
    SHOW_HEAD();
    SHOW_DATA();
    TELL_CELL();
    TELL_FILENAME();
    TELL_OLDDATA();
    TELL_REQUIRE();
    TELL_DATA_CELL();
    READY_FOR_WORKING();
}

/* ----- */

CLEAR_DATA()
{
    for (i=0; i<MREC; ++i) {
        strcpy( data[i], BLANK);
        data[i][SIZE - 2] = 10;
        data[i][SIZE - 1] = 0;
    }
    for (i=0; i<9; ++i)
        data[i][3] = 49 + i;
    for (i=9; i<MREC; ++i) {
        itoa(i+1, cell, 10);
        data[i][2] = cell[0];
        data[i][3] = cell[1];
    }
    strset( filename, 0);
    strcpy( filename, "S" );
    KEY = HOME;
    NINP = 0;
    NBUS = 0;
    ReadData = 1;
    FormZbus = 1;
}

```

```

SaveData = 0;
}

SHOW_HEAD()
{
    clear_screen();
    dumps( 1, 1, HAT_1), dumps( 2, 1, HAT_2), dumps( 3, 1, HAT_3);
    dumps( 5, 1, HAT_5), dumps( 6, 1, HAT_6), dumps( 7, 1, HAT_7);
    lineh( 4, 1, 80, 's'), lineh( 8, 1, 80, 's');
    linev( 5, 1, 23, 's'), linev( 5, 6, 23, 's'), linev( 5, 80, 23, 's');
    dumpc( 4, 1, 218), dumpc( 4, 6, 194), dumpc( 4, 80, 191);
    dumpc( 8, 1, 195), dumpc( 8, 6, 197), dumpc( 8, 80, 180);
    lineh(24, 1, 80, 's');
}

SHOW_DATA()
{
    dump_screen( 9, 3, 23, 4, &data[top][2], SIZE);
    dump_screen( 9, 8, 23, SIZE - 3, &data[top][7], SIZE);
}

TELL_CELL()
{
    char field = fld + 64;
    itoa( rec+1, cell, 10);
    strncat( cell, &field, 1);
    dumps( 1, 8, " ");
    dumps( 1, 8, cell);
}

TELL_FILENAME()
{
    dumps( 1, 40, " ");
    dumps( 1, 40, filename);
}

TELL_OLDDATA()
{
    clear( 2, 10, 2, 20);
    dumpsn( 2, 11, D(rec, fld), LDAT);
}

```

ศูนย์วิทยทรัพยากร

จุฬาลงกรณ์มหาวิทยาลัย

```

}

TELL_OLDDATA_INV()
{
    clear( 2, 10, 2, 20);
    color( 2, 11, REVER, LINV);
    dumpsn( 2, 12, D(rec, fld), LDAT);
}

```

```

TELL_REQUIRE()
{
    clear( 3, 11, 3, 60);
    if (fld == 1 ) dumpsn( 3, 11, REQ_1);
    eif (fld == 2 ) dumpsn( 3, 11, REQ_2);
    eif (fld == 3 ) dumpsn( 3, 11, REQ_3);
    eif (fld == 4 ) {
        if (same(D(rec, 3), "GEN ", 4)) dumpsn( 3, 11, REQ_4a);
        eif (same(D(rec, 3), "XMER", 4)) dumpsn( 3, 11, REQ_4b);
        eif (same(D(rec, 3), "LINE", 4)) dumpsn( 3, 11, REQ_4c);
        else
            dumpsn( 3, 11, "I DON'T KNOW");
    }
    eif (fld == 5 ) dumpsn( 3, 11, REQ_5);
    eif (fld == 6 ) dumpsn( 3, 11, REQ_6);
    eif (fld == 7 ) dumpsn( 3, 11, REQ_7);
    eif (fld == 8 ) dumpsn( 3, 11, REQ_8);
    eif (fld == 9 ) dumpsn( 3, 11, REQ_9);
    eif (fld == 10) dumpsn( 3, 11, REQ_10);
    else
        dumpsn( 3, 11, REQ_11);
}

```

```

TELL_DATA_CELL()
{
    dumpsn( row, CDAT, D(rec, fld), LDAT);
    color( row, CIN, REVER, LINV);
}

```

```

/* ----- */

```

```

READY_FOR_WORKING()
{

```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```

while (KEY != EXIT) {
    if ( F1 < KEY && KEY < F10 ) DO_JOBS();
    if ( F10 == KEY || KEY == '/' ) DO_MENU();
    if ( 31 < KEY && KEY < 127 ) DO_INPUT();
    if ( F10 < KEY && KEY < INS ) DO_MOVE();
    if ( KEY == EXIT ) break;
    READY_STATUS();
    KEY = getkey();
}
ENDING();
}

```

```

READY_STATUS()
{
    dumpsa( 1, 74, " READY ", REVER);
    TELL_READY_STATUS_LINE();
    setcur( row, CDAT+LDAT/2);
}

```

```

ENDING()
{
    clear_line( 25, 25 );
    beep( 'H' );
    exit( 0 );
}

```

```

/*-----*/

```

```

DO_MENU()
{
    color( row, CINV, GREEN, LINV);
    scan_buffer( 1, 1, 25, 80);
    MENU();
    dump_buffer( 1, 1, 25, 80);
    TELL_FILENAME();
    color( row, CINV, REVER, LINV);
}

```

```

DO_MOVE()
{

```

ศูนย์วิทยทรัพยากร

จุฬาลงกรณ์มหาวิทยาลัย



```
color( row, CINV, GREEN, LINV);
if ( MOVE( KEY, MREC, MFLD, 9, 23, &row, &rec, &fld, &top)) SHOW_DATA();
TELL_CELL();
TELL_OLDDATA();
TELL_REQUIRE();
color( row, CINV, REVER, LINV);
}
```

```
DO_INPUT()
{
  dumpsa( 1, 74, " INPUT ", REVER);
  TELL_OLDDATA_INV();
  if ( _scan_input( temp, row, CDAT, LDAT, &KEY))
    if ( INPUT( temp, fld, fldtype, fldlen, fldint, fldmod, D(rec,3)) ) {
      strncpy( D( rec, fld), temp, LDAT );
      for (i=0; i<MREC; ++i)
        if (same( D(i, 1), " ", 5)) break;
      NINP = i;
      ReadData = 1;
      FormZbus = 1;
      SaveData = 1;
    }
  else KEY = 0;
  TELL_OLDDATA();
  TELL_DATA_CELL();
}
```

```
DO_JOBS()
{
  dumpsa( 1, 74, " ", GREEN);
  dumpsa( 1, 75, " JOBS ", REVER);
  switch ( KEY ) {
    case F2 : SAVE_DATA(); break;
    case F6 : PRINT_SCREEN(); break;
  }
  TELL_FILENAME();
}
```

```
/* ----- */
```

```
SAVE_DATA()
{
```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย


```

if ( SaveData ) {
    if (( fx = fopen( filename, "w") ) != NULL) {
        wait("Saving . . .");
        fprintf( fx, "%d\n", NINP);
        for (i=0; i<NINP; ++i)
            fputs( data[i], fx);
        fclose( fx );
        SaveData = 0;
    }
    else error("Incorrect filename");
}
else wait("Saving . . .");
}

LOAD_FILE()
{
    int NR;
    if (scan_subcom("Enter file name for Loading: ", temp, 8, 14, 14, &KEY)) {
        beep('H'), setcur( 1, 80);
        if (( fx = fopen( temp, "r") ) != NULL) {
            fscanf( fx, "%d\n", &NR);
            if (0 < NR && NR <= MREC) {
                wait("Loading DATA . . .");
                CLEAR_DATA();
                NINP = NR;
                for (i=0; i<NINP; ++i)
                    fgets( data[i], SIZE, fx), fscanf( fx, "%d\n", &j);
                fclose( fx );
                strcpy( filename, strupr( temp ));
                ReadData = 1;
                FormZbus = 1;
                LOAD_PRE_RESULTS();
                SaveData = 0;
            }
            else fclose( fx ), error("Number of records in file not match");
        }
        else error("File not found or Disk Drive not Ready.");
    }
    clr_scan_subcom();
}

LOAD_PRE_RESULTS()
{
    READ_DATA_FOR_CALC();
    if ( !ERR ) {
        LOAD_ZBUS( 1 );
        if ( !ERR ) {
            LOAD_ZBUS( 0 );
            if ( !ERR ) {

```



 ศูนย์วิจัยทรัพยากร
 จุฬาลงกรณ์มหาวิทยาลัย


```

LOAD_YP();
if ( !ERR )
    FormZbus = 0;
}
}
}
ERR = 0;
}

```

```

WRITE_FILE()
{
    if (NINP > 0) {
        if (scan_subcom("Enter new file name: ", temp, 8, 11, 14, &KEY)) {
            beep('H'), setcur( 1, 80);
            if (( fx = fopen( temp, "w" )) != NULL) {
                if (confirm("Write to new File ")) {
                    wait("Saving . . .");
                    fprintf( fx, "%d\n", NINP);
                    for (i=0; i<NINP; ++i)
                        fprintf( fx, "%s", data[i]);
                    fclose( fx );
                    SaveData = 0;
                    strcpy( filename, strupr( temp ));
                    KEY = BS;
                }
                else error("Incorected File name");
            }
        }
        clr_scan_subcom();
    }
    else error("Number of input Records = 0, . . . Can't save file");
}

```

```

PRINT_DATA()
(

```

```

    if (( fx = fopen("PRN", "w" )) != NULL) {
        wait("Printing . . . INPUT DATA");
        fprintf( fx, "%c%c", 27, 77);
        fprintf( fx, "File: %s\n", filename);
        fprintf( fx, "%s", HAT_4);
        fprintf( fx, "%s", HAT_5);
        fprintf( fx, "%s", HAT_6);
        fprintf( fx, "%s", HAT_7);
        fprintf( fx, "%s", HAT_8);
        for (i=0; i<NINP; ++i)
            fprintf( fx, "%s", data[i]);
        fprintf( fx, "%s", BOTTOM);
        fprintf( fx, "\n\n" );
    }

```

ศูนย์วิทยทรัพยากร

จุฬาลงกรณ์มหาวิทยาลัย

```

    fclose( fx );
}
else error("Incorrect Printer Name");
}

/* ----- */

```

```

READ_DATA()
{
    wait("Read Input DATA . . . . . Loop count:"), set_count();
    NBUS = 0;
    for (j=1, i=0; i<MREC; ++i, j = i-(i/10)*10) {
        if (j == 0) plus_count();
        T( i, 1), bp[0][i] = bp[1][i] = atoi( temp );
        T( i, 2), bq[0][i] = bq[1][i] = atoi( temp );
        if (bq[1][i] == 0) break;
        Read_Data( i, 3);
        Read_Data( i, 4);
        T( i, 5), zp[1][i].x = atof( temp );
        T( i, 6), zp[1][i].y = atof( temp );
        T( i, 7), zp[0][i].x = atof( temp );
        T( i, 8), zp[0][i].y = atof( temp );
        T( i, 9), zmutu[i].x = atof( temp );
        T( i, 10), zmutu[i].y = atof( temp );
        T( i, 11), lm[i] = atoi( temp );
        NBUS = max( NBUS, max( bp[1][i], bq[1][i] ) );
        if (NBUS > MBUS) errors("Line", i+1, "Too BIG Bus"), ERR = 1;
    }
    NLINE = i;
}

```

```

Read_Data(int r, int f)
{
    if (f == 3)
        if (same( D( r, f), "LINE", 4)) eq[r] = 0;
        eif (same( D( r, f), "GEN ", 4)) eq[r] = 1;
        eif (same( D( r, f), "XMER", 4)) eq[r] = 2;
        else errors("Line", r+1, "Undefined Equipment name"), ERR = 1;
    else
        if ( eq[r] == 0 ) ty[r] = 0;
        eif ( eq[r] == 1 )
            if (same( D( r, f), "YG", 2)) ty[r] = 0;
            eif (same( D( r, f), "Y ", 2)) ty[r] = 3;
            else errors("Line", r+1, "Undefined Generator Type"), ERR = 1;
        else
            if (same( D( r, f), "YG-YG", 5)) ty[r] = 0;

```

ศูนย์วิทยทรัพยากร

จุฬาลงกรณ์มหาวิทยาลัย

```

    eif (same( D( r, f), "D-YG ", 5))    ty[r] = 1;
    eif (same( D( r, f), "YG-D ", 5))    ty[r] = 2;
    eif ( !same( D( r, f), "    ", 4))    ty[r] = 3;
    else errors("Line", r+1, "Undefined Transformer Type"), ERR = 1;
}

```

/* ----- */



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```

/* ----- */
/*
/*      File : MAUX.C
/*
/*      AUXILARY routine for : FA1.EXE & FP1.EXE
/*
/* ----- */

#include <dos.h>
#include <conio.h>
#include <string.h>
#include <stdio.h>

#define eif   else if

#define REVER 120      /* Reverse green color */
#define FLASH 130     /* Blink green color */

static int  VIDEO = 0xB000,
            i, j, x,
            off, count,
            tops, lefts,
            bots, rights;

static char value[7],
            buf[25][80][2];

/* ----- */

SET_VIDEO(char *screen)
{
    if (screen[0] == 'c') VIDEO = 0xB800;
}

/* ----- */

getkey()
{
    int KEY;
    KEY = bioskey(0);
    if ((KEY & 0xFF) == 0) return(256 + (KEY >> 8));
    else return(KEY & 0xFF);
}

```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```

see_more_data(int *kk, int brow)
{
    if (*kk > 22 - brow) {
        pause("Press anykey to see more data.");
        clear_line(brow, 23);
        setcur(brow, 1);    *kk = 0;
    }
    else ++*kk;
}

```

```

swapi(int *ii, int *jj)
{
    x = *ii, *ii = *jj, *jj = x;
}

```

```

pos(char ch, char *string)    /* Position of a char in String */
{
    int position = -1;
    for (j=0; string[j]!=0; ++j)
        if (ch == string[j]) { position = j; break; }
    return(position);
}

```

```

has(char ch, char *string)    /* Has a char. in String or not */
{
    int has;
    char *sear = strchr(string,ch);
    if (*sear == 0) has = 0;
    else has = 1;
    return(has);
}

```

```

have(char *str, char *string)    /* Has string in String or not */
{
    int has;
    char *sear = strstr(string,str);
    if (*sear == 0) has = 0;
    else has = 1;
    return(has);
}

```



คู่มยัวิทยทรพยากร
จุพาลงกรณัมหาวิทยาลัย

```

same(char *s1, char *s2, int n)  /* Is String.1 same String.2 */
{
    int cs, cmp = -55;
    cmp = strncmp(s1, s2, n);
    cs = 0; if (cmp == 0) cs = 1;
    return(cs);
}

```

```

/* ----- */

```

```

setcur(int row, int col)
{
    gotoxy(col, row);
}

```

```

color(int row, int col, char att, int NCHR)
{
    off = 160*row + 2*col - 161;
    for (j=0; j<NCHR; ++j, off += 2)
        pokeb( VIDEO, off, att);
}

```

```

clear(int top, int left, int bot, int right)
{
    x = top*160 + left*2 - 162;
    for (i=top; i<=bot; ++i, x+=160)
        for (off=x, j=left; j<=right; ++j, off+=2)
            pokeb( VIDEO, off, ' '), pokeb( VIDEO, off+1, GREEN);
    setcur( top, left );
}

```

```

clear_screen()
{
    clear( 1, 1, 25, 80);
}

```



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```
clear_line(int top, int bot)
{
    clear( top, 1, bot, 80);
}
```

```
/* ----- */
```

```
dumpc(int row, int col, char ch)
{
    pokeb( VIDEO, row*160 + col*2 - 162, ch);
}
```

```
dumpca(int row, int col, char ch, char att)
{
    off = row*160 + col*2 - 162;
    pokeb( VIDEO, off, ch), pokeb( VIDEO, off+1, att);
}
```

```
dumpsa(int row, int col, char *string)
{
    off = row*160 + col*2 - 162;
    for (j=0; string[j]!=0; ++j, off+=2)
        pokeb( VIDEO, off, string[j]);
}
```

```
dumpsa(int row, int col, char *string, char att)
{
    off = row*160 + col*2 - 162;
    for (j=0; string[j]!=0; ++j, off+=2)
        pokeb( VIDEO, off, string[j]), pokeb( VIDEO, off+1, att);
}
```

```
dumpsn(int row, int col, char *string, int NCHR)
{
    off = row*160 + col*2 - 162;
```

ศูนย์วิทยทรัพยากร

จุฬาลงกรณ์มหาวิทยาลัย



```

for (j=0; j<NCHR; ++j, off+=2)
  pokeb( VIDEO, off, string[j]);
}

```

```

/* ----- */

```

```

beep(char level)

```

```

{
  switch( level ) {
    case 'L': sound(250), delay(250), nosound(); break;
    case 'M': sound(500), delay(125), nosound(); break;
    case 'H': sound(1000), delay(64), nosound(); break;
  }
}

```

```

set_count()

```

```

{
  count = 0;
  itoa(count, value, 10);
  dumpsa( 25, 50, value, LIGHTRED);
}

```

```

plus_count()

```

```

{
  itoa(++count, value, 10);
  dumpsa( 25, 50, value, LIGHTRED);
}

```

```

wait(char *msg)

```

```

{
  beep('H');
  clear_line(25, 25), dumps(25, 1, msg);
  dumpsa(25, 65, "Wait a minute.", FLASH), setcur(25, 80);
}

```

```

confirm(char *msg)

```

```

{

```

ศูนย์วิทยทรัพยากร

จุฬาลงกรณ์มหาวิทยาลัย


```

int key;
beep('M');
clear_line( 25, 25);
dumpsa( 25, 1, msg, LIGHTRED);
setcur( 25, strlen(msg) + 5), printf("Go on ? (Y/N) : ");
key = getkey();
if (key == 'Y' || key == 'y') return(1);
else return(0);
}

```

```

pause(char *msg)
{
    beep('M');
    clear_line(25, 25), dumps(25, 1, msg);
    setcur(25, strlen(msg)+1);
    getkey();
}

```

```

/* ----- */

```

```

linev(int top, int left, int bot, char type)
{
    char line = 0xB3;
    if (type == 'd') line = 0xBA;
    for (i=top; i<=bot; ++i)
        dumpc(i, left, line);
}

```

```

lineh(int top, int left, int right, char type)
{
    char line = 0xC4;
    if (type == 'd') line = 0xCD;
    for (j=left; j<=right; ++j)
        dumpc(top, j, line);
}

```

```

frame(int top, int left, int bot, int right, char type)
{
    char tl = 0xDA, tr = 0xBF, bl = 0xC0, br = 0xD9;
    clear(top, left, bot, right);
}

```

ศูนย์วิทยทรัพยากร

จุฬาลงกรณ์มหาวิทยาลัย

```

if (type == 'd')  t1 = 0xC9,  tr = 0xBB,  b1 = 0xC8,  br = 0xBC;
lineh(top, left+1, right-1, type);
lineh(bot, left+1, right-1, type);
linev(top+1, left, bot-1, type);
linev(top+1, right, bot-1, type);
dumpc(top, left, t1),  dumpc(top, right, tr);
dumpc(bot, left, b1),  dumpc(bot, right, br);
}

```

```

/* ----- */

```

```

scan_buffer(int top, int left, int bot, int right)
{
  x = top*160 + left*2 - 162;
  for (i=top-1; i<bot; ++i, x+=160)
    for (off=x, j=left-1; j<right; ++j, off+=2)
      buf[i][j][0] = peekb( VIDEO, off),
      buf[i][j][1] = peekb( VIDEO, off+1);
}

```

```

dump_buffer(int top, int left, int bot, int right)
{
  x = top*160 + left*2 - 162;
  for (i=top-1; i<bot; ++i, x+=160)
    for (off=x, j=left-1; j<right; ++j, off+=2)
      pokeb( VIDEO, off, buf[i][j][0]),
      pokeb( VIDEO, off+1, buf[i][j][1]);
}

```

```

dump_screen(int top, int left, int bot, int right, char *DATA, int SIZE)
{
  int y;
  --top, --left, x = top*160 + left*2;
  for (i=top; i<bot; ++i, x+=160, DATA += SIZE)
    for (off=x, y=0, j=left; j<right; ++j, ++y, off += 2)
      pokeb( VIDEO, off, DATA[y]);
}

```

```

/* ----- */

```

ศูนย์วิทยทรัพยากร

จุฬาลงกรณ์มหาวิทยาลัย

```

error(char *msg)
{
    beep('L');
    scan_buffer(25, 1, 25, 80);
    clear_line(25, 25);
    dumpsa(25, 1, "!Errors.!", REVER);
    dumpsa(25, 13, msg, LIGHTRED);
    dumps(25, 65, "Press any key."), setcur(25, 80);
    getkey();
    dump_buffer(25, 1, 25, 80);
}

```

```

errors(char *ms, int ii, char *msg)
{
    beep('L');
    scan_buffer(25, 1, 25, 80);
    clear_line(25, 25);
    dumpsa(25, 1, "!Errors.!", REVER);
    setcur(25, 13), printf("%s %d: %s", ms, ii, msg);
    dumps(25, 65, "Press any key."), setcur(25, 80);
    getkey();
    dump_buffer(25, 1, 25, 80);
}

```

/*-----*/

```

scan_input(char *temp, int row, int col, int NCHR, int *KEY)
{
    int SCAN, colf = col - 1,
        cole = col + NCHR;
    clear(row, colf, row, cole);
    dumpc(row, colf, 26);
    dumpc(row, cole, 27);
    SCAN = scan_strings(temp, row, col, NCHR, KEY);
    dumpc(row, colf, 32);
    dumpc(row, cole, 32);
    return(SCAN);
}

```

```

scan_subcom(char *msg, char *temp, int row, int col, int NCHR, int *KEY)

```

```

{
    int NMSG = strlen(msg),
        colb = col + NMSG + 2;
    tops = row;
    lefts = col;
    bots = row + 2;
    rights = col + NMSG + NCHR + 4;
    scan_buffer(tops, lefts, bots, rights);
    frame(tops, lefts, bots, rights, 'd');
    dumpsa(row + 1, col + 2, msg, BROWN);
    return( scan_strings( temp, row + 1, colb, NCHR, KEY) );
}

```

```

clr_scan_subcom()
{
    dump_buffer(tops, lefts, bots, rights);
}

```

```

/* ----- */

```

```

scan_strings(char *temp, int row, int col, int NC, int *KEY)
{
    int MOVE = 1, SCAN = 0, r = row, c = col, N = 0, k = *KEY;
    if (k == 13) k = MOVE = 0;
    setcur(r, c);
    color(r, c, LIGHTRED, NC);
    TELL_SCAN_STATUS_LINE();
    while ( k != 27 ) {
        if (keychar(k))
            if ( N < NC) dumpc(r, c, k), ++c, setcur(r, c), ++N;
            else beep('L');
        if (keyback(k))
            if ( N > 0 ) --c, setcur(r, c), dumpc(r, c, ' '), --N;
            else beep('L');
        if (keymove(k))
            if ( MOVE ) { SCAN = 1; break; }
        if ( k == 13 ) { SCAN = 1; break; }
        if ( k == 320) PRINT_SCREEN();
        k = getkey();
    }
    if (SCAN) scans(temp, row, col, N, MOVE, KEY, k);
    else *KEY = 27;
    return( SCAN );
}

```

```

keychar(int k)
{
    if (31<k && k<127) return(1);
    else return(0);
}

keyback(int k)
{
    if (k==8 || k==127) return(1);
    else return(0);
}

keymove(int k)
{
    if (k==328 || k==331 || k==333 || k==336) return(1);
    else return(0);
}

scans(char *temp, int row, int col, int NCHR, int MOVE, int *KEY, int k)
{
    if (MOVE) *KEY = k;
    else *KEY = 0;
    for (off = row*160 + col*2 - 162, j=0; j<NCHR; ++j, off+=2)
        temp[j] = peekb(VIDEO, off);
    temp[j] = 0;
}

/* ----- ศูนย์วิจัยทรัพยากร ----- */
/* ----- จุฬาลงกรณ์มหาวิทยาลัย ----- */

TELL_READY_STATUS_LINE()
{
    clear_line( 25, 25);
    dumps( 25, 1, "F2-Save F6-Print F10,/--MEMU");
    color( 25, 1, LIGHTCYAN, 2);
    color( 25, 12, LIGHTCYAN, 2);
    color( 25, 24, LIGHTCYAN, 3);
    color( 25, 28, LIGHTCYAN, 1);
}

```

```
TELL_SCAN_STATUS_LINE()
```

```
{
    clear_line( 25, 25);
    dumps( 25, 1, "F6-Print < -Input Esc-Go back");
    color( 25, 1, LIGHTCYAN, 2);
    color( 25, 13, LIGHTCYAN, 1);
    dumpca( 25, 14, 0xC4, LIGHTCYAN);
    dumpca( 25, 15, 0xD9, LIGHTCYAN);
    color( 25, 27, LIGHTCYAN, 3);
}
```

```
TELL_MENU_STATUS_LINE().
```

```
{
    clear_line( 25, 25);
    dumps( 25, 1, "F6-Print < -Do command Esc-Go back");
    color( 25, 1, LIGHTCYAN, 2);
    color( 25, 13, LIGHTCYAN, 1);
    dumpca( 25, 14, 0xC4, LIGHTCYAN);
    dumpca( 25, 15, 0xD9, LIGHTCYAN);
    color( 25, 32, LIGHTCYAN, 3);
}
```

```
PRINT_SCREEN()
```

```
{
    int m, n;
    unsigned char ch;
    for (m=0, i=0; i<25; ++i, m+=160) {
        for (n=0, j=0; j<80; ++j, n+=2) {
            ch = peekb( VIDEO, m+n);
            if (ch == 0 || ch == NULL) ch = 0x20;
            eif (ch == 0xC9 || ch == 0xDA) ch = 0x80;
            eif (ch == 0xBB || ch == 0xBF) ch = 0x81;
            eif (ch == 0xC8 || ch == 0xC0) ch = 0x82;
            eif (ch == 0xBC || ch == 0xD9) ch = 0x83;
            eif (ch == 0xBA || ch == 0xB3) ch = 0x84;
            eif (ch == 0xCD || ch == 0xC4) ch = 0x85;
            eif (ch == 0xCC || ch == 0xC3) ch = 0x86;
            eif (ch == 0xB9 || ch == 0xB4) ch = 0x87;
            eif (ch == 0xCA || ch == 0xC1) ch = 0x88;
            eif (ch == 0xCB || ch == 0xC2) ch = 0x89;
            eif (ch == 0xCE || ch == 0xC5) ch = 0x8A;
            write( 4, &ch, 1);
        }
        ch = 0x0A, write( 4, &ch, 1);
    }
    ch = 0x0A;
    for (i=0; i<5; ++i) write( 4, &ch, 1);
}
```



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```

/* ----- */
/*                                     */
/*      File : MENU.C                 */
/*                                     */
/*      MENU routine for : FA1.EXE    */
/*                                     */
/* ----- */

```

```

#include <bios.h>
#include <string.h>

```

```

#include "KEY.B"

```

```

#define MCOM      6      /* Max. COMmand      */
#define MSUB      7      /* Max. SUB-command  */

#define eif      else if

extern char      filename[15];
extern int      KEY, ERR,
               NBUS,
               ReadData,
               SaveData,
               FormZbus;

static char      temp[15],
               stcom[MCOM][10], /* command string      */
               stsub[MCOM][MSUB][30]; /* sub-command string */
static int      docom, /* do command          */
               dosub[MCOM], /* do sub-command     */
               colcom[MCOM], cM, /* column of command */
               lencom[MCOM], lM, /* length of command  */
               lensub[MCOM], lS, /* length of subcommand */
               nofsub[MCOM], nS; /* number of subcommand */

static int      i, j, k;
static long     time1, time2, time3; /* Begin,End Calc.Time */

int      Type, Phase, Bus=1, Quan, Sol; /* parameter for calc. */
int      MuMemo; /* Mutual Memory used */
long     Memo[2]; /* Memory used */
float    Time[2]; /* Time used Zero, Pos. */

/* ----- */

```

```

SET_MENU()

```

```

{
    strcpy(stcom[0], "Worksheet");
    strcpy(stsub[0][0], "Set Date  ");
    strcpy(stsub[0][1], "Set Time  ");
    strcpy(stsub[0][2], "Clear table");
    strcpy(stcom[1], "File");
}

```



```

strcpy(stsub[1][0], "Load ");
strcpy(stsub[1][1], "Write to");
strcpy(stcom[2], "Calculate");
strcpy(stsub[2][0], "Type of Fault : 3 Phase ");
strcpy(stsub[2][1], "Referece Phase : A ");
strcpy(stsub[2][2], "Fault at Bus : 1 ");
strcpy(stsub[2][3], "Quantities : Sequence");
strcpy(stsub[2][4], "Type of Solution : Partial ");
strcpy(stsub[2][5], "Fault Calculation .");
strcpy(stcom[3], "Display");
strcpy(stsub[3][0], "Data for Positive Seq.");
strcpy(stsub[3][1], "Z-BUS for Positive Seq.");
strcpy(stsub[3][2], "Data for Zero Seq. ");
strcpy(stsub[3][3], "Z-BUS for Zero Seq. ");
strcpy(stsub[3][4], "Fault Solution ");
strcpy(stcom[4], "Print ");
strcpy(stsub[4][0], "Input Data ");
strcpy(stsub[4][1], "Data for Positive Seq.");
strcpy(stsub[4][2], "Z-BUS for Positive Seq.");
strcpy(stsub[4][3], "Data for Zero Seq. ");
strcpy(stsub[4][4], "Z-BUS for Zero Seq. ");
strcpy(stsub[4][5], "Fault Solution ");
strcpy(stcom[5], "Exit");
strcpy(stsub[5][0], "No ");
strcpy(stsub[5][1], "Yes ");
for (i=0; i<MCOM; ++i) {
    for (j=0; stsub[i][j][0] != 0; ++j) ;
    nofsub[i] = j;
}
colcom[i] = colcom[i-1] + lencom[i-1] + 5; /* column com. */
colcom[0] = 3;
lencom[0] = strlen(stcom[0]);
for (i=1; i<MCOM; ++i) {
    colcom[i] = colcom[i-1] + lencom[i-1] + 5; /* column com. */
    lencom[i] = strlen(stcom[i]); /* length com. */
}
for (i=0; i<MCOM; ++i)
    lensub[i] = strlen(stsub[i][0]); /* length sub. */
}
/* ----- */

```



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```

DRAW_MENU()
{
    clear_line(2, 3);
    for (i=0; i<MCOM; ++i) {
        cM = colcom[i];
        dumpsn(3, cM, stcom[i], lencom[i]);
    }
}

```

```

    color(3, cM, LIGHT, 1);
}
COLOR_MENU(1);
}

```

```

COLOR_MENU(int c)
{
    cM = colcom[docom];
    lM = lencom[docom];
    if (c == 0) color(3, cM-1, GREEN, lM+2), color(3, cM, LIGHT, 1);
    else color(3, cM-1, REVER, lM+2);
}

```

```

DRAW_SUB_MENU()
{
    lS = lensub[docom];
    nS = nofsub[docom];
    if (nS > 0) {
        frame(4, cM-2, 5+nS, cM+lS+1, 's');
        for (i=5, j=0; j<nS; ++j, ++i)
            dumpsn(i, cM, stsub[docom][j], lS), color(i, cM, LIGHT, 1);
    }
    COLOR_SUB_MENU(1);
}

```

```

COLOR_SUB_MENU(int c)
{
    k = 5 + dosub[docom];
    if (nS > 0) {
        if (c == 0) color(k, cM-1, GREEN, lS+2), color(k, cM, LIGHT, 1);
        else color(k, cM-1, REVER, lS+2);
    }
}

```

```

CLEAR_SUB_MENU()
{
    dump_buffer(4, cM-2, 5+nS, cM+lS+1);
}

```

```

COM_HOME()

```



ศูนย์บริการสุขภาพ
จุฬาลงกรณ์มหาวิทยาลัย

```

{
  for (i=0; i<MCOM; ++i)
    dosub[i] = 0;
  COLOR_MENU(0);
  CLEAR_SUB_MENU();
  docom = 0;
  COLOR_MENU(1);
  DRAW_SUB_MENU();
}

```

COM_LEFT()

```

{
  COLOR_MENU(0);
  CLEAR_SUB_MENU();
  if (docom < 1) docom = MCOM-1;
  else docom -= 1;
  COLOR_MENU(1);
  DRAW_SUB_MENU();
}

```

COM_RIGHT()

```

{
  COLOR_MENU(0);
  CLEAR_SUB_MENU();
  if (docom == MCOM - 1) docom = 0;
  else docom += 1;
  COLOR_MENU(1);
  DRAW_SUB_MENU();
}

```

SUB_UP()

```

{
  COLOR_SUB_MENU(0);
  if (dosub[docom] < 1) dosub[docom] = nS-1;
  else dosub[docom] -= 1;
  COLOR_SUB_MENU(1);
}

```

SUB_DOWN()

```

{
  COLOR_SUB_MENU(0);
  if (dosub[docom] == nS-1) dosub[docom] = 0;
  else dosub[docom] += 1;
}

```



ศูนย์วิทยทรัพยากร

จุฬาลงกรณ์มหาวิทยาลัย

```

COLOR_SUB_MENU(1);
}

SET_MENU_CURSOR()
{
    setcur( dosub[docom] + 5, cm);
}

/* ----- */

MENU()
{
    dumpsa( 1, 74, " ", GREEN);
    dumpsa( 1, 75, " MENU ", REVER);
    DRAW_MENU();
    DRAW_SUB_MENU();
    TELL_MENU_STATUS_LINE();
    while ( KEY != EXIT ) {
        SET_MENU_CURSOR();
        KEY = getkey();
        if ( KEY == ESC ) { KEY = BS; break; }
        eif ( KEY == HOME ) COM_HOME();
        eif ( KEY == LEFT ) COM_LEFT();
        eif ( KEY == RIGHT ) COM_RIGHT();
        eif ( KEY == UP ) SUB_UP();
        eif ( KEY == DOWN ) SUB_DOWN();
        eif ( KEY == ENTER ) { COMMAND( docom, dosub[docom]); break; }
        eif ( KEY == F6 ) PRINT_SCREEN();
    }
    GOTO_NEXT_SUB();
    if ( KEY != BS && KEY != HOME && KEY != EXIT) MENU();
}

COMMAND(int com, int sub)
{
    if (com == 0)
        if (sub == 0) SET_DATE();
        eif (sub == 1) SET_TIME();
        else CLEAR_DATA();
    eif (com == 1)
        if (sub == 0) LOAD_FILE();
        else WRITE_FILE();
    eif (com == 2)

```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```

if (sub == 0) FAULT_TYPE();
elseif (sub == 1) REFER_PHASE();
elseif (sub == 2) FAULT_AT_BUS();
elseif (sub == 3) SET_QUANTITY();
elseif (sub == 4) SET_SOLUTION();
else
    CALCULATE_FAULT();
endif (com == 3)
if (sub == 0 && !ReadData ) DISPLAY_DATA_SEQ( 1 );
elseif (sub == 1 && !FormZbus ) DISPLAY_ZBUS_SEQ( 1 );
elseif (sub == 2 && !ReadData ) DISPLAY_DATA_SEQ( 0 );
elseif (sub == 3 && !FormZbus ) DISPLAY_ZBUS_SEQ( 0 );
elseif (sub == 4 && !FormZbus ) GOTO_FAULT( 'D' );
else error("Re-arrange Data / Calculate before Display.");
endif (com == 4)
if (sub == 0 ) PRINT_DATA();
elseif (sub == 1 && !ReadData ) PRINT_DATA_SEQ( 1 );
elseif (sub == 2 && !FormZbus ) PRINT_ZBUS_SEQ( 1 );
elseif (sub == 3 && !ReadData ) PRINT_DATA_SEQ( 0 );
elseif (sub == 4 && !FormZbus ) PRINT_ZBUS_SEQ( 0 );
elseif (sub == 5 && !FormZbus ) GOTO_FAULT( 'P' );
else error("Re-arrange Data / Calculate before Printing.");
endif (com == 5)
if (sub == 1) KEY = EXIT;
else
    KEY = BS, beep('H');
}

GOTO_NEXT_SUB()
{
dosub[0] = 0;
dosub[1] = 0;
if (docom == 2) {
    if (dosub[docom] == 5) dosub[docom]=0, dump_buffer(1, 1, 24, 80);
    else
        dosub[docom]++;
}
elseif (docom == 3) {
    if (dosub[docom] == 4) dosub[docom]=0;
    else
        dosub[docom]++;
    dump_buffer(1, 1, 24, 80);
}
elseif (docom == 4) {
    if (dosub[docom] == 5) dosub[docom]=0;
    else
        dosub[docom]++;
}
dosub[5] = 0;
}

/* ----- */

```

```

SET_DATE()
{
    if (scan_subcom( "Enter Date (MM-DD-YY): ", temp, 9, 3, 14, &KEY))
        beep('H');
    clr_scan_subcom();
}

```

```

SET_TIME()
{
    if (scan_subcom( "Enter Time (HH:MM:SS): ", temp, 9, 3, 14, &KEY))
        beep('L');
    clr_scan_subcom();
}

```

```

/* ----- */

```

```

FAULT_TYPE()
{
    clear( 15, 26, 20, 60);
    frame( 15, 26, 20, 60, 's');
    dumps( 16, 28, "1. Three Phase      (3 Phase)");
    dumps( 17, 28, "2. Line to Line      (LL)");
    dumps( 18, 28, "3. Single Line to Ground (SLG)");
    dumps( 19, 28, "4. Double Line to Ground (2LG)");
    if (scan_subcom("Enter Fault No. : ", temp, 12, 26, 2, &KEY))
        if (temp[0] != 0)
            if (has(temp[0], "1234")) {
                Type = temp[0] - 49;
                stsub[2][0][20] = 0;
                if (temp[0] == '1') strcat(stsub[2][0], "3 Phase ");
                eif (temp[0] == '2') strcat(stsub[2][0], "LL ");
                eif (temp[0] == '3') strcat(stsub[2][0], "SLG ");
                else strcat(stsub[2][0], "2LG ");
            }
            else error("Invalid Fault Type");
    dump_buffer( 15, 26, 20, 60);
    clr_scan_subcom();
}

```

```

REFER_PHASE()
{

```

```

if (scan_subcom("Enter Phase ( A, B or C ) : ", temp, 12, 26, 2, &KEY))
  if (temp[0] != 0) {
   strupr(temp);
    if (has(temp[0], "ABC")) {
      Phase = temp[0] - 65;
      stsub[2][1][20] = temp[0];
    }
    else error("Invalid Reference Phase");
  }
clr_scan_subcom();
}

```



```

FAULT_AT_BUS()
{
  if (scan_subcom("Enter Bus No. (0 for ALL) : ", temp, 12, 26, 2, &KEY))
    if (temp[0] != 0) {
      Bus = atoi(temp);
      if (Bus == 0) {
        stsub[2][2][20] = 'A';
        stsub[2][2][21] = 'L';
        stsub[2][2][22] = 'L';
      }
      else {
        stsub[2][2][20] = temp[0];
        stsub[2][2][21] = temp[1];
        stsub[2][2][22] = ' ';
      }
    }
  clr_scan_subcom();
}

```

```

SET_QUANTITY()
{
  clear( 15, 26, 19, 52);
  frame( 15, 26, 19, 52, 's');
  dumps( 16, 28, "1. Sequence (0, 1, 2)");
  dumps( 17, 28, "2. Phase (A, B, C)");
  dumps( 18, 28, "0. Sequence & Phase ");
  if (scan_subcom("Enter Quantity No. : ", temp, 12, 26, 2, &KEY))
    if (temp[0] != 0)
      if (has(temp[0], "012")) {
        Quan = temp[0] - 49;
        stsub[2][3][20] = 0;
        if (temp[0] == '0') strcat(stsub[2][3], "Seq.Pha.");
        eif (temp[0] == '1') strcat(stsub[2][3], "Sequence");
        else strcat(stsub[2][3], "Phase ");
      }
    else error("Invalid Quantity No.");
  dump_buffer( 15, 26, 19, 52);
}

```

ศูนย์วิทยทรัพยากร
มหาวิทยาลัย

```

clr_scan_subcom();
}

SET_SOLUTION()
{
clear( 15, 26, 18, 40);
frame( 15, 26, 18, 40, 's');
dumps( 16, 28, "1. Partial ");
dumps( 17, 28, "2. All    ");
if (scan_subcom("Enter Solution No. : ", temp, 12, 26, 2, &KEY))
    if (temp[0] != 0)
        if (has(temp[0], "12")) {
            Sol = temp[0] - 49;
            stsub[2][4][20] = 0;
            if (temp[0] == '1') strcat(stsub[2][4], "Partial");
            else                strcat(stsub[2][4], "All    ");
        }
        else error("Invalid Solution Type.");
dump_buffer( 15, 26, 18, 40);
clr_scan_subcom();
}

/*-----*/

CALCULATE_FAULT()
{
if ( ReadData ) {
    READ_DATA_FOR_CALC();
}
if ( !ReadData ) {
    if ( FormZbus ) {
        FORM_ZBUS();
        if ( !ERR ) {
            SAVE_DATA();
            SAVE_ZBUS( 1 );
            SAVE_ZBUS( 0 );
            SAVE_YP();
            if ( !ERR )
                FormZbus = 0;
        }
    }
}
if ( !ERR ) GOTO_FAULT( 'D' );
}
}

```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย


```
READ_DATA_FOR_CALC()
```

```
{
  ERR = 0;
  READ_DATA();
  if ( !ERR ) {
    ORDERING_DATA_SEQ( 1 );
    if ( !ERR ) {
      ORDERING_DATA_SEQ( 0 );
      if ( !ERR ) {
        SET_MUTUAL_SEQ( 0 );
        if ( !ERR )
          ReadData = 0;
      }
    }
  }
}
```

```
FORM_ZBUS()
```

```
{
  time1 = biostime(0,0);

  CALC_ZBUS_SEQ( 1 );

  time2 = biostime(0,0);
  Time[1] = (float)(time2 - time1) / 18.2;
  Memo[1] = NBUS * NBUS;

  CALC_ZBUS_SEQ( 0 );

  time3 = biostime(0,0);
  Time[0] = (float)(time3 - time2) / 18.2;
  Memo[0] = Memo[1] + MuMemo;
}
```

```
GOTO_FAULT(char State)
```

```
{
  int b;
  if (Bus == 0) {
    for (Bus=1, b=0; b<NBUS; ++b, ++Bus)
      FAULT_ANA( State );
    Bus = 0;
  }
  elif (Bus <= NBUS)
    FAULT_ANA( State );
  else error("Out off MAX. BUS");
}
```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```
FAULT_ANA(char State)
```

```
{  
  ZBUS_COLUMN();  
  if (Quan == -1) {  
    Quan = 0, FAULT_ANALYSIS( State, So1 );  
    Quan = 1, FAULT_ANALYSIS( State, So1 );  
    Quan = -1;  
  }  
  else    FAULT_ANALYSIS( State, So1 );  
}
```

```
/* ----- */
```



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```

/* ----- */
/*                                     */
/*   File : MOVE.C                   */
/*                                     */
/*   MOVE Routine for : FA1.EXE & FP1.PRJ */
/*                                     */
/* ----- */

```

```
#include "KEY.B"
```

```
#define eif else if
```

```
static int DUMP, NROW,
           MX, MF, BR, ER,
           *r, *x, *f, *t;
```

```
/* ----- */
```

```
MOVE(int KEY,
      int MREC, int MFLD, int BROW, int EROW,
      int *row, int *rec, int *fld, int *top )
```

```

{
  DUMP = 0;
  MX = MREC, MF = MFLD;
  BR = BROW, ER = EROW, NROW = ER-BR+1;
  r = row, x = rec, f = fld, t = top;
  switch(KEY) {
    case HOME : GO_HOME(); break;
    case UP   : MOVE_UP(); break;
    case DOWN : MOVE_DOWN(); break;
    case LEFT : MOVE_LEFT(); break;
    case RIGHT : MOVE_RIGHT(); break;
    case PGUP : MOVE_PAGE_UP(); break;
    case PGDN : MOVE_PAGE_DOWN(); break;
    case END  : GO_TO_BOTTOM(); break;
  }
  if (DUMP) CHECK_TOP(), CHECK_BOTTOM();
  return(DUMP);
}

```

```
/* ----- */
```

```
GO_HOME()
```

```

{
  *r = BR, *x = *t = 0, *f = 1, DUMP = 1;
}

```

```

}

MOVE_UP()

```

```

{
  if (*r > BR)  --*r, --*x;
  eif (*x > 0)  --*x, --*t, DUMP = 1;
}

```

```

MOVE_DOWN()

```

```

{
  if (*r < ER)  ++*r, ++*x;
  eif (*x < MX-1) ++*x, ++*t, DUMP = 1;
}

```

```

MOVE_LEFT()

```

```

{
  if (*f > 1)  --*f;
}

```

```

MOVE_RIGHT()

```

```

{
  if (*f < MF-1) ++*f;
}

```

```

MOVE_PAGE_UP()

```

```

{
  if (*t > NROW-1) *x -= NROW, *t -= NROW, DUMP = 1;
  else GO_HOME();
}

```

```

MOVE_PAGE_DOWN()

```

```

{
  if (*t < MX-NROW*2) *x += NROW, *t += NROW, DUMP = 1;
  else GO_TO_BOTTOM();
}

```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```

GO_TO_BOTTOM()
{
    *r = ER, *x = MX-1, *t = MX-NROW, *f = 1, DUMP = 1;
}

```

```

/* ----- */

```

```

CHECK_TOP()
{
    int CBEG = BR - 1;
    if (*t > 0) {
        dumpc(CBEG, 1, 192);
        dumpc(CBEG, 6, 193);
        dumpc(CBEG, 80, 217);
    }
    else {
        dumpc(CBEG, 1, 195);
        dumpc(CBEG, 6, 197);
        dumpc(CBEG, 80, 180);
    }
}

```

```

CHECK_BOTTOM()
{
    int CBOT = ER + 1;
    if (*t == MX - NROW) {
        dumpc(CBOT, 1, 192);
        dumpc(CBOT, 6, 193);
        dumpc(CBOT, 80, 217);
    }
    else {
        dumpc(CBOT, 1, 196);
        dumpc(CBOT, 6, 196);
        dumpc(CBOT, 80, 196);
    }
}

```

```

/* ----- */

```



ศูนย์วิทยทรัพยากร

มหาวิทยาลัย

```

/* ----- */
/*
/*   File : MPUT.C
/*
/*       INPUT Rountine for : FA1.EXE & FP1.EXE
/*
/* ----- */

#include <ctype.h>
#include <string.h>

#define eif     else if

static char    *D3, *TM;
static int     CANPUT,
              FD, *TY,
              *LD, *LI, *LM,
              i, j, k, ii, kk,
              mm, nn, oo, pp, rr, nmove, nfib;

/* ----- */

INPUT(char *temp, int field, int *type,
      int *lendat, int *lenint, int *lenmod, char *DA3)
{
    CANPUT = 0;
    D3 = DA3;
    TM = temp;
    FD = field;
    TY = type + FD;
    LD = lendat + FD;
    LI = lenint + FD;
    LM = lenmod + FD;
    if (strlen(TM)==1 && TM[0]==' ') {          /* Blank cell */
        for (i=1; i<*LD; ++i)
            TM[i] = ' ';
        CANPUT = 1;
    }
    eif (TM[0] != ' ') {                        /* Input data */
        pp = strlen(TM);
        if (*TY == 1)      CHECK_LABEL();
        else {
            for (mm=j=0; j<pp; ++j)
                if (!isdigit(TM[j]) || TM[j]=='-' || TM[j]=='.') ++mm;
            if (mm == pp) {
                if (*TY == 2) CHECK_INTEGER();
                eif (*TY == 3) CHECK_REAL_A();      /* real + */
                eif (*TY == 4) CHECK_REAL_B();      /* real -/+ */
            }
            else error("Require only Number, Minus sign or Point.");
        }
    }
}

```

```

    }
    else error("Input by begining with blank");
    temp = TM;
    return(CANPUT);
}

```

```

/* ----- */

```

```

CHECK_LABEL()
{
   strupr(TM);
    if (FD == 3) {
        if (have(TM,"GEN XMER LINE")) CANPUT = 1;
        else error("Invalid Equipment");
    }
    eif (FD == 4) {
        if (same(D3,"GEN",3)) {
            if (have(TM,"Y YG")) CANPUT = 1;
            else error("Invalid generator Type");
        }
        eif (same(D3,"XMER",4)) {
            if ( ( have("Y-",TM) !! have("G-",TM) !! have("D-",TM)
                && (have("-Y",TM) !! have("-D",TM))
                && have(TM,"YG-YG YG-Y Y-YG YG-D D-YG Y-D D-Y D-D") )
                CANPUT = 1;
            else error("Invalid transformer Type");
        }
        else error("Required not thing (BLANK)");
    }
    else CANPUT = 1;
    if (CANPUT) ARRANGE_LABEL();
}

```

```

ARRANGE_LABEL()
{
    for (i=pp; i<*LD; ++i)
        TM[i] = ' ';
}

```

```

/* ----- */

```

ศูนย์วิทยทรัพยากร

จุฬาลงกรณ์มหาวิทยาลัย

```

CHECK_INTEGER()
{
    for (mm= j=0; j<pp; ++j)
        if (isdigit(TM[j])) ++mm;
    if (mm == pp) CANPUT = 1;
    else error("Require only Number");
    if (CANPUT) ARRANGE_INTEGER();
}

```

```

ARRANGE_INTEGER()
{
    if (atoi(TM) == 0) {
        for (i=0; i<*LD; ++i)
            TM[i] = ' ';
        TM[*LD-1] = '0';
    }
    else {
        nmove = *LD-pp;
        for (kk=1, k=0; k<nmove; ++k, ++kk)
            TM[*LD-kk] = TM[pp-kk], TM[pp-kk] = ' ';
        for (i=0; i<*LD; ++i)
            if (has(TM[i], "123456789")) break;
            else TM[i] = ' ';
    }
}

```

```

/*-----*/

```

```

CHECK_REAL_A()
{
    for (mm= nn= ii= oo= j=0; j<pp; ++j) {
        if (isdigit(TM[j])) ++mm;
        if (nn > 0) ++oo;
        if (TM[j] == '.') ++mm, ++nn;
        if (nn == 0) ++ii;
    }
    if (mm == pp) {
        if (ii > *LI) error("Out off range.");
        eif (oo > *LM) error("Out off position.");
        eif (nn > 1) error("More than One Point.");
        else CANPUT = 1;
    }
    else error("Require only Number or point.");
    if (CANPUT) {
        if (nn==1 && TM[0]=='.') SHIFT_RIGHT(), TM[0] = '0';
        eif (nn==0 && TM[0]!='.') TM[pp] = '.', ++pp;
        ARRANGE_REAL();
    }
}

```

ศูนย์วิทยุพยาบาล
จุฬาลงกรณ์มหาวิทยาลัย


```

}
}

```

```
SHIFT_RIGHT()
```

```

{
  for (kk=1, k=0; k<pp; ++k, ++kk)
    TM[pp-kk+1] = TM[pp-kk];
  ++pp;
}

```

```
/* ----- */
```

```
CHECK_REAL_B()
```

```

{
  for (mm=nn=ii=oo=rr=j=0; j<pp; ++j) {
    if (isdigit(TM[j])) ++mm;
    if ( nn > 0 ) ++oo;
    if ( TM[j] == '.' ) ++mm, ++nn;
    if ( TM[j] == '-' ) ++mm, ++rr;
    if ( nn == 0 ) ++ii;
  }
  if (MINUS_SIGN()) {
    if (mm == pp) {
      if (ii > *LI) error("Out off range.");
      eif (oo > *LM) error("Set preposition in file TABLE.D again");
      eif (nn > 1) error("More than One Point.");
      eif (nn == 0) TM[pp] = '.', ++pp, CANPUT = 1;
      else CANPUT=1;
    }
    else error("Input Number, Minus sign or Point.");
  }
  if (CANPUT) ARRANGE_REAL();
}

```

```
MINUS_SIGN()
```

```

{
  int sign = 0;
  if (rr > 1) error("Too many minus signs");
  eif (rr == 1) {
    if (TM[0] == '-') {
      if (pp == 1) error("Require some more");
      eif (TM[1] == '.') sign = 1, SHIFT_RIGHT(), TM[0]='-', TM[1]='0', ++mm, ++ii;
      else sign = 1;
    }
    else error("Misplace minus signs");
  }
}

```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



```

}
else {
    if (TM[0] == '.') {
        SHIFT_RIGHT(), TM[0]='0';
        SHIFT_RIGHT(), TM[0]=' ', mm+=2, ii+=2, sign=1;
    }
    else
        SHIFT_RIGHT(), TM[0]=' ', ++mm, ++ii, sign=1;
}
return(sign);
}
}

```

/* ----- */

ARRANGE_REAL()

```

{
    nfib = *LM - oo;
    for (k=0; k<nfib; ++k)
        TM[pp+k] = '0';
    pp += k;
    nmove = *LD - pp;
    for (kk=1, k=0; k<nmove; ++k, ++kk)
        TM[*LD-kk] = TM[pp-kk], TM[pp-kk] = ' ';
}

```

/* ----- */

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```

/* ----- */
/*                                     */
/*      File : COMPLEX.c              */
/*                                     */
/*      COMPLEX founction for : FA1.EXE & FP1.EXE */
/*                                     */
/* ----- */

#include <stdio.h>
#include <math.h>
#define COMPLX struct complex

#define eif      else if

#define P        M_PI_2
#define PI       M_PI
#define PI2      M_PI*2

#define min      1E-4
#define MIN      1E-100
#define MAX      1E+3

/* ----- */

cadd( Z, zF, zB )      COMPLX *Z, zF, zB;      /* Complex add */
{
    Z->x = zF.x + zB.x;
    Z->y = zF.y + zB.y;
    modify_rectan( Z );
}

cstt( Z, zF, zB )      COMPLX *Z, zF, zB;      /* Com.subtract */
{
    Z->x = zF.x - zB.x;
    Z->y = zF.y - zB.y;
    modify_rectan( Z );
}

cmul( Z, zF, zB )      COMPLX *Z, zF, zB;      /* Com.multiple */
{
    polar( &zF );
    polar( &zB );
    Z->x = zF.x * zB.x;
    Z->y = zF.y + zB.y;
}

```

ศูนย์วิทยทรัพยากร

จุฬาลงกรณ์มหาวิทยาลัย

```

    modify_polar( Z );
    rectan( Z );
}

```

```

cdiv( Z, zF, zB )      COMPLEX *Z, zF, zB;      /* Comp. divide */
{
    polar( &zF );
    polar( &zB );
    if ( zB.x > MIN ) Z->x = zF.x / zB.x;
    else               Z->x = MAX , printf("\nDevide by 0. ");
    Z->y = zF.y - zB.y;
    modify_polar( Z );
    rectan( Z );
}

```

```

/* ----- */

```

```

polar( COMPLEX *Z )      /* rectan to Polar in Radius */
{
    COMPLEX T = *Z;
    Z->x = cabs( T );
    if ( T.x < -MIN ) Z->y = PI + atan( T.y / T.x );
    eif ( T.x > MIN ) Z->y = atan( T.y / T.x );
    else
        if ( T.y < -MIN ) Z->y = -P;
        eif ( T.y > MIN ) Z->y = P;
        else Z->y = 0;
    if ( Z->x < MIN ) Z->x = 0;
}

```

```

modify_polar( COMPLEX *Z )
{
    if ( Z->x < MIN ) Z->x = 0;
    if ( Z->y > PI2 ) Z->y -= PI2;
    eif ( Z->y < -PI2 ) Z->y += PI2;
}

```

```

rectan( COMPLEX *Z )
{
    COMPLEX U = *Z;

```

ศูนย์วิทยทรัพยากร

จุฬาลงกรณ์มหาวิทยาลัย

```

Z->x = U.x * cos( U.y );
Z->y = U.x * sin( U.y );
modify_rectan( Z );
}

```

```

modify_rectan( COMPLEX *Z )
{
  if ( fabs( Z->x ) < MIN ) Z->x = 0;
  if ( fabs( Z->y ) < MIN ) Z->y = 0;
}

```

```

/* ----- */

```

```

polar( COMPLEX *Z )          /* rectan to Polar in Degree */
{
  polar( Z );
  if ( Z->x < min )           Z->y = 0;
  else {                     Z->y += 180/PI;
    if ( Z->y > 180 )         Z->y -= 360;
    eif ( Z->y < -180 )      Z->y += 360;
  }
}

```

```

/* ----- */

```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```

/* ----- */
/*                                     */
/*      File : N1_ORDER.c             */
/*                                     */
/*      ORDERING data routine for : FA1.EXE */
/*                                     */
/* ----- */

```

```

#include <stdio.h>
#include <string.h>
#include <math.h>
#define COMPLEX struct complex

```

```

#include "HAT.B"
#include "MAX.B"

```

```

#define eif else if

```

```

extern int ERR,
           NBUS,
           NLINE,
           ty[MREC],
           Tm[MREC],
           bp[2][MREC],
           bq[2][MREC];
extern COMPLEX zp[2][MREC],
              zmutu[MREC];

```

```

static FILE *fx;

```

```

static int i, j, k,
           S, NCHK, NUSE,
           L, R, NL;

```

```

char BusList[2][MBUS+1],
     LineList[2][MREC+1];
int use[MREC];

```

```

/* ----- */

```

```

ORDERING_DATA_SEQ(int Sequence)

```

```

{
    S = Sequence;
    if ( S == 0 ) CONVERT_DATA();
    CLEAR_ORDERING();
    COLLECT_GEN();
    COLLECT_LINE();
    if ( S == 0 ) ORDERING_MUTUAL();
}

```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```

CONVERT_DATA()
{
    for (i=0; i<NLINE; ++i) {
        if ( ty[i] == 1 ) bp[S][i] = 0;
        eif ( ty[i] == 2 ) bq[S][i] = bp[S][i], bp[S][i] = 0;
    }
}

```

```

/* ----- */

```

```

CLEAR_ORDERING()
{
    strset( BusList[S], 0 );
    strset( LineList[S], 0 );
    for (i=0; i<NLINE; ++i) use[i] = 0;
}

```

```

COLLECT_GEN()
{
    NUSE = 0;
    for (i=0; i<NLINE; ++i) {
        if ( S == 0 && ty[i] == 3) use[i] = 1, ++NUSE;
        eif ( bp[S][i] == 0 ) Collect_Data( i, bq[S][i] );
    }
}

```

```

Collect_Data(int r, char Q)
{
    char L;
    use[r] = 1, ++NUSE;
    L = r + 1, strcat( LineList[S], &L, 1);
    if ( !has( Q, BusList[S] ) ) strcat( BusList[S], &Q, 1);
}

```

```

COLLECT_LINE()
{
    while ( NUSE < NLINE ) {
        NCHK = NUSE;

```

ศูนย์วิทยทรัพยากร

จุฬาลงกรณ์มหาวิทยาลัย


```

DISPLAY_DATA_POSITIVE()

```

```

{
    NL = NLINE;
    clear_line( 1, 23 );
    printf( DAT_10          );
    printf( DAT_11          );
    printf( DAT_12 , NL, NBUS );
    printf( DAT_13          );
    printf( DAT_14          );
    printf( DAT_13          );
    k = 0;
    for (i=0; i<NL; ++i) {
        L = LineList[1][i], R = L - 1;
        printf( FORM_1, L, bp[1][R], bq[1][R], zp[1][R] );
        see_more_data( &k, 7);
    }
    pause( "press any key to go on" );
}

```

```

DISPLAY_DATA_ZERO()

```

```

{
    NL = strlen( LineList[0] );
    clear_line( 1, 23 );
    printf( DAT_00          );
    printf( DAT_01          );
    printf( DAT_02 , NL, NBUS );
    printf( DAT_03          );
    printf( DAT_04          );
    printf( DAT_03          );
    k = 0;
    for (i=0; i<NL; ++i) {
        L = LineList[0][i], R = L - 1;
        if ( lm[R] == 0 )
            printf( FORM_1, L, bp[0][R], bq[0][R], zp[0][R] );
        else
            printf( FORM_2, L, bp[0][R], bq[0][R], zp[0][R], zmtu[R], lm[R] );
        see_more_data( &k, 7);
    }
    pause( "press any key to go on" );
}

```

```

/* ----- */

```

```

PRINT_DATA_SEQ(int Sequence)
{
    S = Sequence;
    if ( S == 0 ) PRINT_DATA_ZERO();
    else          PRINT_DATA_POSITIVE();
}

```

```

PRINT_DATA_POSITIVE()
{
    if ( (fx = fopen( "PRN","w" ) ) != NULL ) {
        wait( "Printing . . . DATA FOR POSITIVE SEQUENCE" );
        NL = NLINE;
        fprintf( fx, "%c%c", 27, 77 );
        fprintf( fx, DAT_10 );
        fprintf( fx, DAT_11 );
        fprintf( fx, DAT_12 , NL, NBUS );
        fprintf( fx, DAT_13 );
        fprintf( fx, DAT_14 );
        fprintf( fx, DAT_13 );
        for (i=0; i<NL; ++i) {
            L = LineList[1][i], R = L - 1;
            fprintf( fx, FORM_1, L, bp[1][R], bq[1][R], zp[1][R] );
        }
        fprintf( fx, DAT_13 );
        fprintf( fx, "\n\n" );
        fclose( fx );
    }
    else error( "Incorrect Printer Name." );
}

```

```

PRINT_DATA_ZERO()
{
    if ( (fx = fopen( "PRN","w" ) ) != NULL ) {
        wait( "Printing . . . DATA FOR ZERO SEQUENCE" );
        NL = strlen( LineList[0] );
        fprintf( fx, "%c%c", 27, 77 );
        fprintf( fx, DAT_00 );
        fprintf( fx, DAT_01 );
        fprintf( fx, DAT_02 , NL, NBUS );
        fprintf( fx, DAT_03 );
        fprintf( fx, DAT_04 );
        fprintf( fx, DAT_03 );
        for (i=0; i<NL; ++i) {
            L = LineList[0][i], R = L - 1;
            if ( lm[R] == 0 )
                fprintf( fx, FORM_1, L, bp[0][R], bq[0][R], zp[0][R] );
        }
    }
}

```

ศูนย์วิทยทรัพยากร

จุฬาลงกรณ์มหาวิทยาลัย

```
    else  
        fprintf( fx, FORM_2, L, bp[0][R], bq[0][R], zp[0][R],zmutu[R],lm[R]);  
    }  
    fprintf( fx, DAT_03 );  
    fprintf( fx, "\n\n" );  
    fclose( fx );  
}  
else    error( "Incorrect Printer Name." );  
}
```

/* ----- */



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```

/* ----- */
/*                                     */
/*   File : N2_MUTUAL.c               */
/*                                     */
/*   SET MUTUAL GROUP routine for : FA1.EXE */
/*                                     */
/* ----- */

#include <stdio.h>
#include <string.h>
#include <math.h>
#define COMPLEX struct complex

#include "MAX.B"

#define eif else if
#define MIN 1E-20

extern char filename[15],
BusList[2][MREC+1],
LineList[2][MREC+1];

extern int ERR,
MuMemo,
Im[MREC],
bp[2][MREC],
bq[2][MREC];

extern COMPLEX zp[2][MREC],
zmutu[MREC];

static char fileYP[15];
static int i, j, k,
X, L, LM, E;

static FILE *fx;

static COMPLEX ZI[MI][MI*2]; /* Z Inverse */

/* Output */
char MuLine[MI][MI+1]; /* mutual line */
int mp[MI][MI], /* mutual bus P */
mq[MI][MI], /* mutual bus Q */
mpp[MI][MI], /* mutual bus P */
mqq[MI][MI], /* mutual bus Q */
EL[MI], /* End Line in mu.group */
NG; /* Number of mu.Groups */

COMPLEX C00 = {0,0},
C10 = {1,0},
Zbuf, Zsum,
YP[MI][MI][MI]; /* YP Impedence */

/* ----- */

```

```

SET_MUTUAL_SEQ(int S)
{
    int NL = strlen( LineList[S] );
    wait( "Calculating . . . MUTUAL          Loop count:" ), set_count();
    for (NG= X=0; X<NL; ++X) {
        plus_count();
        L = LineList[0][X];
        LM = 1m[L-1];
        if ( LM > 0 ) SET_MUTUAL();
    }
    END_MUTUAL();
    CALC_MU_MEMO();
}

```

```

SET_MUTUAL()
{
    int LL, LLL;
    for (E= i=0; i<X; ++i) {
        plus_count();
        LL = LineList[0][i];
        if ( LM == LL ) {
            fill_data( LL );
            for (j=0; j<X; ++j) {
                plus_count();
                LLL = LineList[0][j];
                if ( LL == 1m[LLL-1] ) fill_data( LLL );
            }
        }
    }
    if (E > 0) {
        EL[NG] = E;
        fill_data( L );
        MuLine[NG][E] = 0;
        CREAT_ZI(), CREAT_YP();
        ++NG;
    }
}

```

```

fill_data(int LINE)
{
    MuLine[NG][E] = LINE;
    mp[NG][E] = bp[0][LINE-1];
    mq[NG][E] = bq[0][LINE-1];
    mpp[NG][E] = bp[1][LINE-1];
    mqq[NG][E] = bq[1][LINE-1];
    ++E;
}

```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```
/* ----- */
```

```
CREAT_ZI()
```

```
{
  int LN, R;
  for (i=0; i<E; ++i) {
    LN = MuLine[NG][i];
    for (j=0; j<E; ++j) {
      plus_count();
      R = MuLine[NG][j] - 1;
      if ( i == j ) YP[NG][i][j] = zp[0][R];
      eif ( LN == 1m[R] ) YP[NG][i][j] = zmutu[R];
      else YP[NG][i][j] = COO;
    }
  }
  for (i=0; i<E; ++i)
    for (j=0; j<E; ++j)
      YP[NG][j][i] = YP[NG][i][j];
}
```

```
CREAT_YP()
```

```
{
  for (i=0; i<E; ++i) {
    plus_count();
    for (j=0; j<E; ++j) {
      ZI[i][j] = YP[NG][i][j];
      if (cabs( ZI[i][j] ) <= MIN ) ZI[i][j] = COO;
      if (j == i) ZI[i][j+E] = C10;
      else ZI[i][j+E] = COO;
    }
  }
```

```
INVERSE_ZI(E);
```

```
plus_count();
for (i=0; i<E; ++i)
  for (j=0; j<E; ++j)
    YP[NG][i][j] = ZI[i][j+E];
}
```

```
INVERSE_ZI(int DIM)
```

```
{
  int BREAK = 0, TDIM = DIM*2;
```



```

if (muP0 == muP1 || muQ0 == muQ1) ;
eif (muP0 == muQ1) Swap_mubus_group( i, 1);
eif (muP1 == muQ0) {
    posP0 = pos( muP0, BusList[0] );
    posP1 = pos( muP1, BusList[0] );
    posQ0 = pos( muQ0, BusList[0] );
    posQ1 = pos( muQ1, BusList[0] );
    if ( posP0 > posQ0 ) Swap_mubus_group( i, 0);
    eif ( posP1 > posQ1 ) Swap_mubus_group( i, 1);
    else error( "Mutual O.K. But Ordering Not O.K." ), ERR = 1;
}
else error( "Mutual Not O.K." ), ERR = 1;
}
}
}
}

```

```

Swap_mubus_group(int g, int x)
{
    int R;
    swapi( &mp[g][x], &mq[g][x] );
    R = MuLine[g][x] - 1;
    swapi( &bp[0][R], &bq[0][R] );
}

```

/* ----- */

```

CALC_MU_MEMO()
{
    plus_count();
    MuMemo = 0;
    for (i=0; i<NG; ++i)
        MuMemo += ((EL[i] + 1) * (EL[i] + 1));
}

```

/* ----- */

SAVE_YP()

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย


```

int NN, MULN;
strset( fileYP, 0 );
strcpy( fileYP, filename );
strcat( fileYP, "Y" );
if (( fx = fopen( fileYP, "w" )) != NULL ) {
    wait( "Saving YP file . . ." );
    fprintf( fx, "Xd\n", NG);
    for (i=0; i<NG; ++i) {
        fprintf( fx, "X3d\n", EL[i] );
        NN = EL[i] + 1;
        for (j=0; j<NN; ++j) {
            MULN = MuLine[i][j];
            fprintf( fx, " X3d X3d X3d", MULN, mpp[i][j], mqq[i][j] );
            for (k=0; k<NN; ++k)
                fprintf( fx, " X#9.5f X#9.5f", YP[i][j][k] );
            fprintf( fx, "\n" );
        }
    }
    fprintf( fx, "Xd\n", MuMemo );
    fclose( fx );
}
else error( "Drive not Ready." ), ERR =1;
}

```

LOAD_YP()

```

{
int NN, MULN;
double vX, vY;
strset( fileYP, 0 );
strcpy( fileYP, filename );
strcat( fileYP, "Y" );
if (( fx = fopen( fileYP, "r" )) != NULL ) {
    wait( "Loading YP file . . ." );
    fscanf( fx, "Xd\n", &NG);
    for (i=0; i<NG; ++i) {
        fscanf( fx, "Xd\n", &EL[i] );
        NN = EL[i] + 1;
        for (j=0; j<NN; ++j) {
            fscanf( fx, "XdXdXd", &MULN, &mpp[i][j], &mqq[i][j] );
            MuLine[i][j] = MULN;
            for (k=0; k<NN; ++k) {
                fscanf( fx, "X1fX1f", &vX, &vY);
                YP[i][j][k].x = vX;
                YP[i][j][k].y = vY;
            }
            fscanf( fx, "\n" );
        }
        MuLine[i][j] = 0;
    }
    fscanf( fx, "Xd\n", &MuMemo );
}

```

```
fclose( fx );  
}  
else ERR =1;  
}
```

/* ----- */



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```

/* ----- */
/*
/*      File : N3_ZBUS.c
/*
/*      ZBUS Calculating routine for : FA1.EXE
/*
/* ----- */

#include <stdio.h>
#include <alloc.h>
#include <stdlib.h>
#include <string.h>
#include <math.h>
#define COMPLEX struct complex

#include "HAT.B"
#include "MAX.B"

#define eif else if

extern char filename[15],
BusList[2][MBUS+1], /* ORDE */
LineList[2][MREC+1],
MuLine[MI][MI+1]; /* MUTU */

extern int ERR,
Bus,
NBUS,
NLINE,
mp[MI][MI],
mq[MI][MI],
EL[MI], NG,
bp[2][MREC],
bq[2][MREC];

extern COMPLEX zp[2][MREC],
C00, C10,
Zbuf, Zsum,
YP[MI][MI][MI];

extern float Time[2];
extern long Memo[2];

static FILE *fx;
static char fileZ[15];

static int i, j, k,
S, P, Q, L, NL, /* Sequence, bus P,Q,L */
G, E, mP, mQ, /* mu Group, Side & Bus */
Cbus[MBUS+1], /* count meet bus Q */
FORMULA; /* formula for calc. */

static COMPLEX Zppq, /* Z in step */
Znear;

COMPLEX far *Z[2][MBUS+1]; /* Zbus matrix */
COMPLEX far *ZBC[2]; /* Zbus Column */

```

```

long size = 16 * ( MBUS+1 );      /* Zbus matrix */

/* ----- */

SET_FAR()
{
  for (i=0; i<2; ++i) {
    for (j=0; j<=MBUS; ++j) {
      Z[i][j] = farmalloc( size );
      if ( Z[i][j] == NULL ) error( "Not enough room for Data" ), exit(1 );
    }
  }
  for (i=0; i<2; ++i)
    for (j=0; j<=MBUS; ++j)
      for (k=0; k<=MBUS; ++k)
        Z[i][j][k].x = Z[i][j][k].y = 0;
  for (i=0; i<2; ++i) {
    ZBC[i] = farmalloc( size );
    if ( ZBC[i] == NULL ) error( "Not enough room for Data" ), exit(1 );
  }
  for (i=0; i<2; ++i)
    for (j=0; j<=MBUS; ++j)
      ZBC[i][j].x = ZBC[i][j].y = 0;
}

/* ----- */

```

ศูนย์วิทยทรัพยากร

CALC_ZBUS_SEQ(int Sequence)

```

{
  int x, line, rec, bP, bQ, H = 0;
  S = Sequence;
  NL = strlen( LineList[S] );
  if ( S == 0 ) wait( "Calculating . . . ZBUS SEQ. 0 Loop count:" );
  else wait( "Calculating . . . ZBUS SEQ. 1 Loop count:" );
  for (i=0; i<NBUS; ++i)
    Cbus[i] = 0;
  set_count();
  for (x=0; x<NL; ++x) {
    line = LineList[S][x];          /* line used */
    rec = line - 1;                /* record used */
    bP = bp[S][rec];              /* bus P */
  }
}

```

```

bQ = bq[S][rec];          /* bus Q      */
P = pos( bP, BusList[S]); /* position P */
Q = pos( bQ, BusList[S]); /* position Q */
H = max(max( H, P),max( H, Q)); /* High bus */
L = H + 1;                /* position L */
Zpqpq = zp[S][rec];      /* Z line    */
if ( S == 1 )             SET_FORMULA( 0, bP);
else {
    if (!MUTUAL( line )) SET_FORMULA( 0, bP);
    else                  SET_FORMULA( 1, bP);
}
CREATE_ZBUS_BY_FORMULA();
}
SORT_ZBUS();
}

```

```

SET_FORMULA(int Mutual, int BP)
{
    if ( !Mutual )
        if ( ++Cbus[Q] < 2 )
            if ( BP == 0 ) FORMULA = 1;
            else          FORMULA = 2;
        else
            if ( BP == 0 ) FORMULA = 3;
            else          FORMULA = 4;
    else
        if ( ++Cbus[Q] < 2 )
            if ( BP == 0 ) FORMULA = 5;
            else          FORMULA = 6;
        else
            if ( BP == 0 ) FORMULA = 7;
            else          FORMULA = 8;
}

```

```

MUTUAL(int LINE)

```

```

{
    int PLUS_YP = 0;
    for (i=0; i<NG; ++i) {
        E = EL[i];
        if (LINE == MuLine[i][E]) {
            PLUS_YP = 1, G = i;
            break;
        }
    }
    return( PLUS_YP );
}

```

ศูนย์วิทยทรัพยากร

จุฬาลงกรณ์มหาวิทยาลัย

```
CREATE_ZBUS_BY_FORMULA()
```

```
{
  switch ( FORMULA ) {
    case 1: CALC_FORMULA_1_(); break;
    case 2: CALC_FORMULA_2_(); break;
    case 3: CALC_FORMULA_3_(); break;
    case 4: CALC_FORMULA_4_(); break;
    case 5: CALC_FORMULA_5_(); break;
    case 6: CALC_FORMULA_6_(); break;
    case 7: CALC_FORMULA_7_(); break;
    case 8: CALC_FORMULA_8_(); break;
  }
}
```

```
CALC_FORMULA_1_()
```

```
{
  for (i=0; i<L; ++i) {
    plus_count();
    if (i != Q)
      Z[S][Q][i] = Z[S][i][Q] = C00;
  }
  Z[S][Q][Q] = Zpqpq;
}
```

```
CALC_FORMULA_2_()
```

```
{
  for (i=0; i<L; ++i) {
    plus_count();
    if (i != Q)
      Z[S][Q][i] = Z[S][i][Q] = Z[S][P][i];
  }
  cadd( &Znear, Z[S][P][Q], Zpqpq ), Z[S][Q][Q] = Znear;
}
```

```
CALC_FORMULA_3_()
```

```
{
  for (i=0; i<L; ++i) {
    plus_count();
    cstt( &Znear, C00, Z[S][Q][i] );
    Z[S][i][L] = Z[S][L][i] = Znear;
  }
  cstt( &Znear, Zpqpq, Z[S][Q][L] ), Z[S][L][L] = Znear;
  ELIMINATE_BUS_L();
}
```

```

CALC_FORMULA_4_()
{
  for (i=0; i<L; ++i) {
    plus_count();
    cstt( &Znear, Z[S][P][i], Z[S][Q][i] );
    Z[S][i][L] = Z[S][L][i] = Znear;
  }
  cstt( &Zbuf, Z[S][P][L], Z[S][Q][L] );
  cadd( &Znear, Zbuf, Zpqq ), Z[S][L][L] = Znear;
  ELIMINATE_BUS_L();
}

```

```

CALC_FORMULA_5_()
{
  for (i=0; i<L; ++i) {
    plus_count();
    if (i != Q) {
      for ( Zsum = C00, j=0; j<E; ++j) {
        mP = pos( mp[G][j], BusList[S] );
        mQ = pos( mq[G][j], BusList[S] );
        cstt( &Zbuf, Z[S][mP][i], Z[S][mQ][i] );
        cmul( &Zbuf, YP[G][E][j], Zbuf );
        cadd( &Zsum, Zsum, Zbuf );
      }
      cdiv( &Znear, Zsum, YP[G][E][E] );
      Z[S][i][Q] = Z[S][Q][i] = Znear;
    }
  }
  for ( Zsum = C00, j=0; j<E; ++j) {
    plus_count();
    mP = pos( mp[G][j], BusList[S] );
    mQ = pos( mq[G][j], BusList[S] );
    cstt( &Zbuf, Z[S][mP][Q], Z[S][mQ][Q] );
    cmul( &Zbuf, YP[G][E][j], Zbuf );
    cadd( &Zsum, Zsum, Zbuf );
  }
  cadd( &Zsum, Zsum, C10 );
  cdiv( &Znear, Zsum, YP[G][E][E] ), Z[S][Q][Q] = Znear;
}

```

```

CALC_FORMULA_6_()
{
  for (i=0; i<L; ++i) {
    plus_count();
    if (i != Q) {
      for ( Zsum = C00, j=0; j<E; ++j) {

```

```

    mP = pos( mp[G][j], BusList[S]);
    mQ = pos( mq[G][j], BusList[S]);
    cstt( &Zbuf, Z[S][mP][i], Z[S][mQ][i] );
    cmul( &Zbuf, YP[G][E][j], Zbuf );
    cadd( &Zsum, Zsum, Zbuf );
}
cdiv( &Zsum, Zsum, YP[G][E][E] );
cadd( &Znear, Z[S][P][i], Zsum );
Z[S][i][Q] = Z[S][Q][i] = Znear;
}
}
for ( Zsum = C00, j=0; j<E; ++j) {
    plus_count();
    mP = pos( mp[G][j], BusList[S]);
    mQ = pos( mq[G][j], BusList[S]);
    cstt( &Zbuf, Z[S][mP][Q], Z[S][mQ][Q] );
    cmul( &Zbuf, YP[G][E][j], Zbuf );
    cadd( &Zsum, Zsum, Zbuf );
}
cadd( &Zsum, Zsum, C10 );
cdiv( &Zsum, Zsum, YP[G][E][E] );
cadd( &Znear, Z[S][P][Q], Zsum ), Z[S][Q][Q] = Znear;
}

CALC_FORMULA_7_(
{
    for (i=0; i<L; ++i) {
        plus_count();
        for ( Zsum = C00, j=0; j<E; ++j) {
            mP = pos( mp[G][j], BusList[S]);
            mQ = pos( mq[G][j], BusList[S]);
            cstt( &Zbuf, Z[S][mP][i], Z[S][mQ][i] );
            cmul( &Zbuf, YP[G][E][j], Zbuf );
            cadd( &Zsum, Zsum, Zbuf );
        }
        cdiv( &Zsum, Zsum, YP[G][E][E] );
        cstt( &Znear, Zsum, Z[S][Q][i] );
        Z[S][i][L] = Z[S][L][i] = Znear;
    }
    for ( Zsum = C00, j=0; j<E; ++j) {
        plus_count();
        mP = pos( mp[G][j], BusList[S]);
        mQ = pos( mq[G][j], BusList[S]);
        cstt( &Zbuf, Z[S][mP][L], Z[S][mQ][L] );
        cmul( &Zbuf, YP[G][E][j], Zbuf );
        cadd( &Zsum, Zsum, Zbuf );
    }
    cadd( &Zsum, Zsum, C10 );
    cdiv( &Zsum, Zsum, YP[G][E][E] );
    cstt( &Znear, Zsum, Z[S][Q][L] ), Z[S][L][L] = Znear;
    ELIMINATE_BUS_L();
}
}

```



```

CALC_FORMULA_8_()
{
  for (i=0; i<L; ++i) {
    plus_count();
    for ( Zsum = C00, j=0; j<E; ++j) {
      mP = pos( mp[G][j], BusList[S]);
      mQ = pos( mq[G][j], BusList[S]);
      cstt( &Zbuf, Z[S][mP][i], Z[S][mQ][i] );
      cmul( &Zbuf, YP[G][E][j], Zbuf );
      cadd( &Zsum, Zsum, Zbuf );
    }
    cdiv( &Zsum, Zsum, YP[G][E][E] );
    cstt( &Zsum, Zsum, Z[S][Q][i] );
    cadd( &Znear, Z[S][P][i], Zsum );
    Z[S][i][L] = Z[S][L][i] = Znear;
  }
  for ( Zsum = C00, j=0; j<E; ++j) {
    plus_count();
    mP = pos( mp[G][j], BusList[S]);
    mQ = pos( mq[G][j], BusList[S]);
    cstt( &Zbuf, Z[S][mP][L], Z[S][mQ][L] );
    cmul( &Zbuf, YP[G][E][j], Zbuf );
    cadd( &Zsum, Zsum, Zbuf );
  }
  cadd( &Zsum, Zsum, C10 );
  cdiv( &Zsum, Zsum, YP[G][E][E] );
  cstt( &Zsum, Zsum, Z[S][Q][L] );
  cadd( &Znear, Zsum, Z[S][P][L] ), Z[S][L][L] = Znear;
  ELIMINATE_BUS_L();
}

```

```

ELIMINATE_BUS_L()
{
  for (i=0; i<L; ++i) {
    for (j=0; j<L; ++j) {
      plus_count();
      cdiv( &Zbuf, Z[S][i][L], Z[S][L][L] );
      cmul( &Zbuf, Zbuf, Z[S][L][j] );
      cstt( &Znear, Z[S][i][j], Zbuf ), Z[S][i][j] = Znear;
    }
  }
}

```

```

SORT_ZBUS()
{

```

```

int I, J, R, C;
for (J=1, j=0; j<NBUS; ++j, ++J) {
    plus_count();
    for (I=1, i=0; i<NBUS; ++i, ++I) {
        R = pos( I, BusList[S]);
        ZBC[S][I] = Z[S][R][j];
    }
    for (I=1, i=0; i<NBUS; ++i, ++I) {
        Z[S][i][j] = ZBC[S][I];
    }
}
for (I=1, i=0; i<NBUS; ++i, ++I) {
    plus_count();
    for (J=1, j=0; j<NBUS; ++j, ++J) {
        C = pos( J, BusList[S]);
        ZBC[S][J] = Z[S][i][C];
    }
    for (J=1, j=0; j<NBUS; ++j, ++J)
        Z[S][i][j] = ZBC[S][J];
}
}
}

```

```

ZBUS_COLUMN()
{
    int I, C = Bus - 1;
    plus_count();
    for (I=1, i=0; i<NBUS; ++i, ++I) {
        ZBC[0][I] = Z[0][i][C];
        ZBC[1][I] = Z[1][i][C];
    }
}

```

ศูนย์วิทยทรัพยากร

จุฬาลงกรณ์มหาวิทยาลัย

```

LOAD_ZBUS(int Sequence)
{
    int I, J;
    double vX, vY;
    S = Sequence;
    strset( fileZ, 0 );
    strcpy( fileZ, filename );
    if ( S == 1 ) strcat( fileZ, "P" );
    if ( S == 0 ) strcat( fileZ, "Z" );
}

```

```

if (( fx = fopen( fileZ, "r" )) != NULL ) {
  if ( S == 1 ) wait( "Loading ZBUS SEQ. 1 file . . ." );
  if ( S == 0 ) wait( "Loading ZBUS SEQ. 0 file . . ." );
  fscanf( fx, "%d\n", &NBUS);
  for (i=0; i<NBUS; ++i)
    for (j=i; j<NBUS; ++j) {
      fscanf( fx, "%d%d%1f%1f\n", &I, &J, &vX, &vY);
      Z[S][i][j].x = vX;
      Z[S][i][j].y = vY;
    }
  fscanf( fx, "%f\n", &Time[S] );
  fscanf( fx, "%d\n", &Memo[S] );
  fclose( fx );
  for (i=0; i<NBUS; ++i)
    for (j=i; j<NBUS; ++j)
      Z[S][j][i] = Z[S][i][j];
}
else ERR = 1;
}

SAVE_ZBUS(int Sequence)
{
  int I, J;
  S = Sequence;
  strset( fileZ, 0 );
  strcpy( fileZ, filename );
  if ( S == 1 ) strcat( fileZ, "P" );
  if ( S == 0 ) strcat( fileZ, "Z" );
  if (( fx = fopen( fileZ, "w" )) != NULL ) {
    if ( S == 1 ) wait( "Saving ZBUS SEQ. 1 file . . ." );
    if ( S == 0 ) wait( "Saving ZBUS SEQ. 0 file . . ." );
    fprintf( fx, "%3d\n", NBUS);
    for (I=1, i=0; i<NBUS; ++i, ++I)
      for (J=I, j=i; j<NBUS; ++j, ++J)
        fprintf( fx, "%3d %3d %20.17f %20.17f\n", I, J, Z[S][i][j] );
    fprintf( fx, "%f\n", Time[S] );
    fprintf( fx, "%d\n", Memo[S] );
    fclose( fx );
  }
  else error( "Disk Drive not ready." ), ERR = 1;
}

```

```

/* ----- */

```

```

DISPLAY_ZBUS_SEQ(int Sequence)
{
    int I, J;
    double Vx, Vy,
           min = 0.00001;
    S = Sequence;
    clear_line( 1, 23);
           printf( ZB_00 );
    if ( S == 0 ) printf( ZB_10 );
    else         printf( ZB_11 );
           printf( ZB_20 );
           printf( ZB_30 );

    k = 0;
    for (I=1, i=0; i<NBUS; ++i, ++I) {
        for (J=I, j=i; j<NBUS; ++j, ++J) {
            Vx = fabs( Z[S][i][j].x );
            Vy = fabs( Z[S][i][j].y );
            if ( Vx < min && Vy < min ) printf( FORM_3, I, J );
            eif ( Vx < min )             printf( FORM_4, I, J, Z[S][i][j].y );
            eif ( Vy < min )             printf( FORM_5, I, J, Z[S][i][j].x );
            else                          printf( FORM_6, I, J, Z[S][i][j] );
            see_more_data( &k, 7);
        }
    }
    printf( "\n" );
    printf( "-----\n" );
    printf( "Time used   = %7.1f Seconds\n", Time[S] );
    printf( "Memory used = %7d Units\n",   Memo[S] );
    printf( "-----\n" );
    pause( "press any key to go on" );
}

/* ----- */

```

```

PRINT_ZBUS_SEQ(int Sequence)
{
    int I, J;
    double Vx, Vy,
           min = 0.00001;

    S = Sequence;
    if (( fx = fopen( "PRN","w" )) != NULL ) {
        wait( "Printing . . . ZBUS COLUMN" );
        fprintf( fx, "%c%c", 27, 77);
           fprintf( fx, ZB_00 );
        if ( S == 0 ) fprintf( fx, ZB_10 );
        else         fprintf( fx, ZB_11 );
           fprintf( fx, ZB_20 );
           fprintf( fx, ZB_30 );
    }
}

```

ศูนย์วิทยทรัพยากร

จุฬาลงกรณ์มหาวิทยาลัย

```

for (I=1, i=0; i<NBUS; ++i, ++I) {
  for (J=I, j=i; j<NBUS; ++j, ++J) {
    Vx = fabs( Z[S][i][j].x );
    Vy = fabs( Z[S][i][j].y );
    if ( Vx < min && Vy < min ) fprintf( fx, FORM_3, I, J );
    eif ( Vx < min ) fprintf( fx, FORM_4, I, J, Z[S][i][j].y );
    eif ( Vy < min ) fprintf( fx, FORM_5, I, J, Z[S][i][j].x );
    else fprintf( fx, FORM_6, I, J, Z[S][i][j] );
  }
}
fprintf( fx, "\n" );
fprintf( fx, "-----\n" );
fprintf( fx, "Time used = %7.1f Seconds\n", Time[S] );
fprintf( fx, "Memory used = %7d Units\n", Memo[S] );
fprintf( fx, "-----\n" );
fprintf( fx, "%c", 12);
fclose( fx );
}
else error( "Incorrect Printer Name." );
}

/* ----- */

```



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



```
/* ----- */
/*
/*   File : N4_FAULT.c
/*
/*   FAULT Calculation routine for : FA1.EXE
/*
/* ----- */

#include <stdio.h>
#include <string.h>
#include <math.h>
#define COMPLEX struct complex

#include "HAT.B"
#include "MAX.B"

#define eif else if
#define A 0
#define B 1
#define C 2

extern char MuLine[MI][MI+1];
extern int Type, Phase, Bus, Quan,
ERR,
NBUS,
NLINE,
EL[MI], NG,
eq[MREC],
ty[MREC],
bp[2][MREC],
bq[2][MREC],
mpp[MI][MI],
mqq[MI][MI];

extern COMPLEX zp[2][MREC],
YP[MI][MI][MI],
C00, C10;

extern COMPLEX far *ZBC[2];

static FILE *fx;

static char buslink[MBUS+1];
static int Sol,
i, j, k,
x, I, P,
bP, bQ,
pP, pQ,
mP, mQ,
NBL, NLL,
MUSIZE, G, X;

static COMPLEX BOT, FRONT,
DV, BACK, SM,
C20 = { 2, 0},
C30 = { 3, 0},
Ep = { 1, 0}, /* pre-fault voltage; bus fault */
```

```

Ei = { 1, 0},          /* pre-fault voltage; other bus */
zF = { 0, 0},          /* fault impedance                */
zg = { 0, 0},          /* ground impedance                */
A1 = {-.5, .866025},   /* temporary Complex variable     */
A2 = {-.5,-.866025};  /* temporary Complex variable     */

char  BusLink[MBUS+1]; /* Bus Links                        */
int   bp1[MREC],       /* bus P in buslink                */
      bq1[MREC];       /* bus Q in buslink                */
COMPLX IB[3],          /* I Bus fault (Sequence)          */
        EB[3][MBUS],  /* V Bus fault (Sequence)          */
        IL[3][MREC+MLAP]; /* I Line fault (Sequence)        */
/* ----- */

```

```

FAULT_ANALYSIS(char state, int sol)
{
  wait( "Calculating . . . FAULT SOLUTION   Loop count:" ), set_count();
  P = Bus;
  Sol = sol;
  if ( Sol == 0 ) SET_BUSLINK();
  else           GET_ALL_BUS();
  BUS_CURRENT();
  BUS_VOLTAGE();
  LINE_CURRENT();
  if ( IERR ) {
    if ( Quan == 1 ) TRANSFER_TO_PHASE_QUANTITY();
    POLAR_OF_SOLUTION();
    if ( state == 'D' ) DISPLAY_FAULT_SOLUTION();
    eif ( state == 'P' ) PRINT_FAULT_SOLUTION();
  }
}

```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```

GET_ALL_BUS()
{
  for (I=1, i=0; i<NBUS; ++i, ++I)
    BusLink[i] = I;
  BusLink[i] = 0;
}

```

```

SET_BUSLINK()
{
    strset( buslink, 0 );
    for (i=0; i<NLINE; ++i) {
        if ( bp[1][i] == Bus ;; bq[1][i] == Bus ) {
            collect_buslink( bp[1][i], bq[1][i] );
            if ( LINE_MUTUAL( i+1 ) ) {
                for (j=0; j<MUSIZE; ++j) { /* Mutual */
                    mP = mpp[G][j], mQ = mqq[G][j];
                    collect_buslink( mP, mQ );
                }
            }
        }
    }
    Sort_BusLink();
}

```

```

collect_buslink(char BP, char BQ)
{
    if ( BP == 0 ) {
        if ( !has( BQ, buslink ) ) strcat( buslink, &BQ, 1 );
    }
    else {
        if ( !has( BP, buslink ) ) strcat( buslink, &BP, 1 );
        if ( !has( BQ, buslink ) ) strcat( buslink, &BQ, 1 );
    }
}

```

```

Sort_BusLink()
{
    char I;
    strset( BusLink, 0 );
    for (I=1, i=0; i<NBUS; ++i, ++I)
        if ( has( I, buslink ) )
            strcat( BusLink, &I, 1 );
}

```

```

/* ----- */

```

```

BUS_CURRENT()
{
    plus_count();
}

```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย


```

if ( Type == 0 ) THREE_PHASE();
elseif ( Type == 1 ) LINE_TO_LINE();
elseif ( Type == 2 ) SINGLE_LINE_TO_GROUND();
else
    DOUBLE_LINE_TO_GROUND();
}

```

```

THREE_PHASE()
{
    IB[0] = IB[2] = C00;          /* Ip(F)(0 ) & Ip(F)(2) */
    cadd( &BOT, zF, ZBC[1][P] );
    cdiv( &IB[1], Ep, BOT );     /* Ip(F)(1) */
}

```

```

LINE_TO_LINE()
{
    IB[0] = C00;          /* Ip(F)(0) */
    cmul( &FRONT, C20, ZBC[1][P] );
    cadd( &BOT, FRONT, zF );
    cdiv( &IB[1], Ep, BOT );     /* Ip(F)(1) */
    cstt( &IB[2], C00, IB[1] ); /* Ip(F)(2) */
    if (Phase == 1) {          /* ref.phase B */
        cmul( &IB[2], A1, IB[1] );
        cstt( &IB[2], C00, IB[2] );
    }
    if (Phase == 2) {        /* ref.phase C */
        cmul( &IB[2], A2, IB[1] );
        cstt( &IB[2], C00, IB[2] );
    }
}

```

```

SINGLE_LINE_TO_GROUND()
{
    cmul( &BACK, C20, ZBC[1][P] );
    cadd( &FRONT, ZBC[0][P], BACK );
    cadd( &BOT, FRONT, zF );
    cdiv( &IB[0], Ep, BOT );     /* IpF0 */
    IB[1] = IB[2] = IB[0];     /* IpF1 , IpF2 */
    if (Phase == 1) {          /* ref.phase B */
        cmul( &IB[0], A2, IB[0] );
        cmul( &IB[1], A1, IB[0] );
        cmul( &IB[2], A2, IB[0] );
    }
    if (Phase == 2) {        /* ref.phase C */
        cmul( &IB[0], A1, IB[0] );
        cmul( &IB[1], A2, IB[0] );
        cmul( &IB[2], A1, IB[0] );
    }
}

```

ศูนย์วิทยทรัพยากร

จุฬาลงกรณ์มหาวิทยาลัย

```

}
}

DOUBLE_LINE_TO_GROUND()
(
  COMPLEX T1,T2,T3,T4, S1,S2,S3, MM,DD, Z00,Z11,Z22, Zdet;
  cadd( &T1, ZBC[0][P], ZBC[1][P] );
  cmul( &T2, C20, zF );
  cmul( &T3, C30, zg );
  cadd( &S1, T1, T2 ); cadd( &S1, S1, T3 );
  cadd( &T4, ZBC[0][P], zF );
  cadd( &S2, T4, T3 );
  cadd( &S3, ZBC[1][P], zF );
  if (Phase == 0) { /* ref.phase A */
    cmul( &MM, S2, S3 );
    cdiv( &DD, MM, S1 );
    cadd( &MM, S3, DD ); cdiv( &IB[1], Ep, MM ); /* IpF1 */
    cdiv( &DD, S2, S1 ); cmul( &IB[2], DD, IB[1] ); /* IpF2 */
    cstt( &IB[2], C00, IB[2] );
    cdiv( &DD, S3, S1 ); cmul( &IB[0], DD, IB[1] ); /* IpF0 */
    cstt( &IB[0], C00, IB[0] );
  }
  else {
    Z00 = S1;
    cmul( &MM, C20, ZBC[1][P] );
    cadd( &Z11, MM, T2 );
    Z22 = S3;
    cmul( &MM, Z00, Z11 );
    cmul( &DD, Z22, Z2 );
    cstt( &Zdet, MM, DD );
    cdiv( &T4, Ep, Zdet );
    if ( Phase == 1 ) { /* ref.phase B */
      cmul( &MM, A2, Z22 );
      cmul( &IB[0], MM, T4 );
      cstt( &IB[0], C00, IB[0] );
      cmul( &IB[1], Z00, T4 );
      cmul( &MM, A2, IB[0] );
      cmul( &DD, A1, IB[1] );
      cadd( &IB[2], MM, DD );
      cstt( &IB[2], C00, IB[2] );
    }
    else { /* ref.phase C */
      cmul( &MM, A1, Z22 );
      cmul( &IB[0], MM, T4 );
      cstt( &IB[0], C00, IB[0] );
      cmul( &IB[1], Z00, T4 );
      cmul( &MM, A1, IB[0] );
      cmul( &DD, A2, IB[1] );
      cadd( &IB[2], MM, DD );
      cstt( &IB[2], C00, IB[2] );
    }
  }
}
)

```

}

/* ----- */

BUS_VOLTAGE()

```

{
  COMPLEX AEi = Ei;
  if ( Type == 2 ) {
    if ( Phase == B ) cmul( &AEi, A2, Ei );
    eif ( Phase == C ) cmul( &AEi, A1, Ei );
  }
  k = 1;
  for (I=1, i=0; i<NBUS; ++i, ++I) {
    if ( has( I, BusLink ) ) {
      plus_count();
      cmul( &BACK, IB[0], ZBC[0][I] ), cstt( &EB[0][k], COO, BACK );
      cmul( &BACK, IB[1], ZBC[1][I] ), cstt( &EB[1][k], AEi, BACK );
      cmul( &BACK, IB[2], ZBC[1][I] ), cstt( &EB[2][k], COO, BACK );
      ++k;
    }
  }
  NBL = k;
}

```

/* ----- */

LINE_CURRENT()

```

{
  int pmP, pmQ, swapPQ;
  COMPLEX TERM;
  k = 0;
  for (i=0; i<NLINE; ++i) {
    swapPQ = 0;
    bP = bp[1][i], bQ = bq[1][i];
    if ( bP == P || bQ == P || So1 == 1 ) {
      plus_count();
      if ( bP == P )
        swapPQ = bP, bP = bQ, bQ = swapPQ, swapPQ = 1;

      pP = pos( bP, BusLink ) + 1;
      pQ = pos( bQ, BusLink ) + 1;
      if ( bP == 0 ) pP = 0;
    }
  }
}

```

ศูนย์วิทยทรัพยากร

จุฬาลงกรณ์มหาวิทยาลัย

```

if (eq[i] == 1) cstt( &DV, C10, EB[1][pQ] ); /* IL(1) */
else          cstt( &DV, EB[1][pP], EB[1][pQ] ); /* IL(1) */
cdiv( &IL[1][k], DV, zp[1][i] );

cstt( &DV, EB[2][pP], EB[2][pQ] ); /* IL(2) */
cdiv( &IL[2][k], DV, zp[1][i] );

if ( LINE_MUTUAL( i+1 ) ) { /* IL(0) */
    for (SM = C00, j=0; j<MUSIZE; ++j) { /* Mutual */
        plus_count();
        mP = mpp[G][j], mQ = mqq[G][j];
        if ( swapPQ )
            mP = mqq[G][j], mQ = mpp[G][j];
        pmP = pos( mP, BusLink ) + 1;
        pmQ = pos( mQ, BusLink ) + 1;
        cstt( &DV, EB[0][pmP], EB[0][pmQ] );
        cmul( &TERM, YP[G][X][j], DV), cadd( &SM, SM, TERM );
    }
    IL[0][k] = SM; /* IL(0) */
}
else { /* No Mutual */
    if ( ty[i] == 0 ) {
        cstt( &DV, EB[0][pP], EB[0][pQ] );
        cdiv( &IL[0][k], DV, zp[0][i] ); /* IL(0) */
    }
    else
        IL[0][k] = C00;
}

bp1[k] = bP, bq1[k] = bQ;
++k;
}
}
NLL = k;
if ( Type > 1 ) APPEND_LINE_CURRENT();
}

```

```

APPEND_LINE_CURRENT()
{

```

```

    k = NLL;
    for (i=0; i<NLINE; ++i) {
        if ( eq[i] == 2 && bq[0][i] == P ) {
            if ( ty[i] == 1 || ty[i] == 2 ) {
                plus_count();
                bQ = bq[0][i];
                pQ = pos( bQ, BusLink ) + 1;
                cstt( &DV, EB[0][0], EB[0][pQ] );
                cdiv( &IL[0][k], DV, zp[0][i] ); /* IL(0) */
                IL[1][k] = IL[2][k] = C00; /* IL(1),(2) */
                bp1[k] = 0, bq1[k] = bQ;
            }
        }
    }
}

```

ศูนย์วิทยทรัพยากร

จุฬาลงกรณ์มหาวิทยาลัย

```

        ++k;
    }
}
NLL = k;
if (NLL > MREC+MLAP) error( "Too many converted line" ), ERR = 1;
}

```

```

LINE_MUTUAL(int Line)

```

```

{
    int group, member, mutual = 0;
    if ( Type > 1 ) {
        for (group=0; group<NG; ++group) {
            member = pos( Line, MuLine[ group ] );
            if ( member > -1 ) {
                G = group;
                X = member;
                MUSIZE = EL[G] + 1;
                mutual = 1;
            }
        }
    }
    return( mutual );
}

```

```

/* ----- */

```

```

TRANSFER_TO_PHASE_QUANTITY()

```

```

{
    COMPLEX TO, T1, T2, TA, TB;
    plus_count();
    TO = IB[0], T1 = IB[1], T2 = IB[2];
    cadd( &SM, TO, T1 ), cadd( &SM, SM, T2 ), IB[A] = SM;
    cmul( &TA, A2, T1 ), cmul( &TB, A1, T2 );
    cadd( &SM, TO, TA ), cadd( &SM, SM, TB ), IB[B] = SM;
    cmul( &TA, A1, T1 ), cmul( &TB, A2, T2 );
    cadd( &SM, TO, TA ), cadd( &SM, SM, TB ), IB[C] = SM;
    for (I=1; I<NBL; ++I) {
        plus_count();
        TO = EB[0][I], T1 = EB[1][I], T2 = EB[2][I];
        cadd( &SM, TO, T1 ), cadd( &SM, SM, T2 ), EB[A][I] = SM;
        cmul( &TA, A2, T1 ), cmul( &TB, A1, T2 );
        cadd( &SM, TO, TA ), cadd( &SM, SM, TB ), EB[B][I] = SM;
        cmul( &TA, A1, T1 ), cmul( &TB, A2, T2 );
        cadd( &SM, TO, TA ), cadd( &SM, SM, TB ), EB[C][I] = SM;
    }
}

```

```

for (x=0; x<NLL; ++x) {
    plus_count();
    T0 = IL[0][x], T1 = IL[1][x], T2 = IL[2][x];
    cadd( &SM, T0, T1 ), cadd( &SM, SM, T2 ), IL[A][x] = SM;
    cmul( &TA, A2, T1 ), cmul( &TB, A1, T2 );
    cadd( &SM, T0, TA ), cadd( &SM, SM, TB ), IL[B][x] = SM;
    cmul( &TA, A1, T1 ), cmul( &TB, A2, T2 );
    cadd( &SM, T0, TA ), cadd( &SM, SM, TB ), IL[C][x] = SM;
}
}

```

```

/*-----*/

```

```

POLAR_OF_SOLUTION()
{
    plus_count();
    polard( &IB[0] );
    polard( &IB[1] );
    polard( &IB[2] );
    for (I=1; I<NBL; ++I) {
        plus_count();
        polard( &EB[0][I] );
        polard( &EB[1][I] );
        polard( &EB[2][I] );
    }
    for (x=0; x<NLL; ++x) {
        plus_count();
        polard( &IL[0][x] );
        polard( &IL[1][x] );
        polard( &IL[2][x] );
    }
}

```

```

/*-----*/

```

```

DISPLAY_FAULT_SOLUTION()
{
    DISPLAY_OUTPUT();
    DISPLAY_BUS_CURRECT();
    DISPLAY_BUS_VOLTAGE();
    DISPLAY_LINE_CURRENT();
}

```

```

DISPLAY_OUTPUT()
{
    clear_line( 1, 23 );
    printf( OUT___ );
    if ( Type == 0 ) printf( OUT_00 , P, Phase+65 );
    eif ( Type == 1 ) printf( OUT_01 , P, Phase+65 );
    eif ( Type == 2 ) printf( OUT_02 , P, Phase+65 );
    else
        printf( OUT_03 , P, Phase+65 );
    printf( OUT___ );
    if ( Quan == 0 ) printf( "\nSEQUENCE QUANTITIES\n" );
    else
        printf( "\nPHASE QUANTITIES\n" );
}

DISPLAY_BUS_CURRENT()
{
    printf( "\nBUS CURRENT\n" );
    if ( Quan == 0 ) printf( OUT_1 );
    else
        printf( OUT_1P );
    printf( FORM_7 , P, IB[0], IB[1], IB[2] );
}

DISPLAY_BUS_VOLTAGE()
{
    printf( "\nBUS VOLTAGE\n" );
    if ( Quan == 0 ) printf( OUT_2 );
    else
        printf( OUT_2P );
    k = 0;
    for ( I=1; I<NBL; ++I ) {
        printf( FORM_7 , BusLink[I-1], EB[0][I], EB[1][I], EB[2][I] );
        see_more_data( &k, 13 );
    }
    if ( NBL + NLL > 8 ) {
        pause( "press anykey to go on" );
        clear_line( 6, 23 );
    }
}

DISPLAY_LINE_CURRENT()
{
    printf( "\nLINE CURRENT\n" );
    if ( Quan == 0 ) printf( OUT_3 );
    else
        printf( OUT_3P );
    k = 0;
}

```

```

PRINT_BUS_VOLTAGE()
{
    wait( "Printing . . . BUS VOLTAGE" );
        fprintf( fx, "\nBUS VOLTAGE\n" );
    if ( Quan == 0 ) fprintf( fx, OUT_2 );
    else fprintf( fx, OUT_2P );
    for (I=1; I<NBL; ++I)
        fprintf( fx, FORM_7 , BusLink[I-1], EB[0][I], EB[1][I], EB[2][I] );
}

PRINT_LINE_CURRENT()
{
    wait( "Printing . . . LINE CURRENT" );
        fprintf( fx, "\nLINE CURRENT\n" );
    if ( Quan == 0 ) fprintf( fx, OUT_3 );
    else fprintf( fx, OUT_3P );
    for (x=0; x<NLL; ++x)
        fprintf( fx, FORM_8 , bp1[x], bq1[x], IL[0][x], IL[1][x], IL[2][x] );
    fprintf( fx, "\n" );
    fprintf( fx, OUT___ );
    fprintf( fx, "\n\n" );
}

/* ----- */

```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



โปรแกรมการวิเคราะห์การลัดวงจร โดยวิธีแยกเป็นส่วนย่อย



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```

/* ----- */
/*                                     */
/*      File : FP.a                    */
/*                                     */
/*      Individaul file for :  FP.EXE  */
/*                                     */
/*      TITLE routine for :  FP1.EXE   */
/*                                     */
/* ----- */

#include <stdio.h>
#include <conio.h>
#include <process.h>

char  Title[ ] =
"
"
"  <----->
"
"      DEPARTMENT OF ELECTRICAL ENGINEERING
"
"      FACULTY OF ENGINEERING
"
"      CHULALONGKORN UNIVERSITY
"
"  {-----}
"
"      COMPUTER AIDED
"
"      FAULT ANALYSIS BY PIECEWISE SOLUTION METHOD
"
"  {-----}
"
"      BY
"
"      THANA CHATAVARAHA  M.ENG. (E.E.)
"
"  [-----]
"
"Program Loading      Please wait . . ."
;

main()
{
    int i;
    struct text_info monotor;
    gettextinfo( &monotor );
    textmode( monotor.currmode );
    for (i=0; Title[i] > 0; ++i)
        switch ( Title[i] ) {
            case '<': Title[i] = 201; break;
            case '>': Title[i] = 187; break;
        }
}

```

ศูนย์วิทยทรัพยากร

จุฬาลงกรณ์มหาวิทยาลัย

```

case '{': Title[i] = 199; break;
case '}': Title[i] = 182; break;
case '[': Title[i] = 200; break;
case ']': Title[i] = 188; break;
case '=': Title[i] = 205; break;
case '-': Title[i] = 196; break;
case '|': Title[i] = 186; break;
}
textattr( GREEN + ( BLACK << 4 ) ), cputs( Title );
if (monotor.currmode == 7)  execl( "FP1.EXE", "FP1.EXE", "mono", NULL);
else
    execl( "FP1.EXE", "FP1.EXE", "color", NULL);
}
/*-----*/

```



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

MAIN2.OBJ
MAUX.OBJ
MOVE.OBJ
MPUT.OBJ
MENU2.OBJ
NO_PLEX.OBJ
N1_ZONE.OBJ
N2_Z124.OBJ
P3_ZBUSEC.OBJ
P4_FAULT.OBJ



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```

/* ----- */
/*                                     */
/*   File: FORM.B   -->   Format Definition   */
/*                                     */
/* ----- */

#define A      0
#define B      1
#define C      2

#define DATA_01 "          DATA FOR ZERO SEQUENCE\n"
#define DATA_02 "          -----\n"
#define DATA_03 "Number of Lines =%3d          Number of Buses =%3d\n"
#define DATA_04 "          -----\n"
#define DATA_05 "LINE      BUS  BUS          ZERO          MUTUAL          LINE\n"
#define FORM_0  " %3d      %3d  %3d      %8.4f %8.4fj      %8.4f %8.4fj      %3d\n"

#define DATA_11 "          DATA FOR POSITIVE SEQUENCE\n"
#define DATA_12 "          -----\n"
#define DATA_13 "Number of Lines =%3d          Number of Buses =%3d\n"
#define DATA_14 "          -----\n"
#define DATA_15 "LINE      BUS  BUS          POSITIVE\n"
#define FORM_1  " %3d      %3d  %3d      %8.4f %8.4fj\n"

#define ZBUS__1 "          -----\n"
#define ZBUS_02 "          ZBUS FOR ZERO SEQUENCE          \n"
#define ZBUS_12 "          ZBUS FOR POSITIVE SEQUENCE          \n"
#define ZBUS__3 "          -----\n"
#define ZBUS_04 "BUS      BUS          R          X          \n"

#define FORM_2  "%3d      %3d          0.0          0.0\n"
#define FORM_3  "%3d      %3d          0.0          %9.5f\n"
#define FORM_4  "%3d      %3d          %9.5f          0.0\n"
#define FORM_5  "%3d      %3d          %9.5f          %9.5f\n"

#define FORM_6  "%3d      %3d -%3d      %9.5f %9.5fj\n"
#define FORM_7  "%3d -%3d      %3d -%3d      %9.5f %9.5fj\n"

#define OUT_00  " BALANCE THREE PHASE FAULT AT BUS %3d REFERENCE PHASE : %c \n"
#define OUT_01  " LINE TO LINE FAULT AT BUS %3d REFERENCE PHASE : %c \n"
#define OUT_02  " SINGLE LINE TO GROUND FAULT AT BUS %3d REFERENCE PHASE : %c \n"
#define OUT_03  " DOUBLE LINE TO GROUND FAULT AT BUS %3d REFERENCE PHASE : %c \n"
#define OUT__  "          -----\n"

#define OUT_1  "Bus          I0      ANGO          I1      ANG1          I2      ANG2\n"
#define OUT_1P "Bus          IA      ANGA          IB      ANGB          IC      ANGC\n"

#define OUT_2  "Bus          V0      ANGO          V1      ANG1          V2      ANG2\n"
#define OUT_2P "Bus          VA      ANGA          VB      ANGB          VC      ANGC\n"

#define OUT_3  "Bus to bus          I0      ANGO          I1      ANG1          I2      ANG2\n"
#define OUT_3P "Bus to bus          IA      ANGA          IB      ANGB          IC      ANGC\n"

```

```
#define FORM_8  "x3d          x#8.4f x#8.2f      x#8.4f x#8.2f      x#8.4f x#8.2f\n"
#define FORM_9  "x3d      x3d      x#8.4f x#8.2f      x#8.4f x#8.2f      x#8.4f x#8.2f\n"
```

/* ----- */



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```

/* ----- */
/*                                     */
/*   File: HAT.B   -->   Head Table Definition   */
/*                                     */
/* ----- */

```

```

                                     File:
#define HAT_1 "Cell: 1A
#define HAT_2 "OldData: Space
#define HAT_3 "Require: Pressing Key
#define HAT_4 "-----\n
#define HAT_5 "   :   Bus           Self Impedance           Mutual Impedance ;\n
#define HAT_6 "   |Line| ----- Equip. Type -----\n
#define HAT_7 "   :   P Q           Positive           Zero           Zm           Line ;\n
#define HAT_8 "   |-----|\n
#define BLANK "   :   |\n
#define BOTTOM "-----\n

#define REQ_1 "NUMBER; 0 to 49"
#define REQ_2 "NUMBER; 1 to 50"
#define REQ_3 "GEN, XMER or LINE"
#define REQ_4a "Y or YG"
#define REQ_4b "YG-YG, YG-Y, Y-YG, YG-D, D-YG, Y-D, D-Y or D-D"
#define REQ_4c " ( BLANK / SPACE ) "
#define REQ_5 "VALUE; 0 to 9.9999"
#define REQ_6 "VALUE; -9.9999 to 9.9999"
#define REQ_7 "VALUE; 0 to 9.9999"
#define REQ_8 "VALUE; -9.9999 to 9.9999"
#define REQ_9 "VALUE; 0 to 9.9999"
#define REQ_10 "VALUE; -9.9999 to 9.9999"
#define REQ_11 "NUMBER; 1 to 99"

/* ----- */

```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```

/* ----- */
/*
/*   File: KEY.B   -->   Keyboard Definition
/*
/* ----- */

#define BS      8      /* Back space */
#define TAB     9      /* Tab        */
#define ENTER  13     /* Enter/Return */
#define ESC    27     /* Esc (Escape) */

#define F1     315    /* Function key */
#define F2     316
#define F3     317
#define F4     318
#define F5     319
#define F6     320
#define F7     321
#define F8     322
#define F9     323
#define F10    324


#define HOME   327    /* Move key */
#define UP     328
#define PGUP   329
#define LEFT   331
#define RIGHT  333
#define END    335
#define DOWN   336
#define PGDN   337

#define INS    338    /* Insert key */
#define DEL    339    /* Delete key */
#define EXIT   350    /* Ctrl F1 Program Exit */

#define GREEN  2      /* normal Green color */
#define LIGHT  10     /* Light green color */
#define REVER  120    /* Reverse green color */
#define FLASH  130    /* Flashing green color */

/* ----- */

```



 วิทยาลัยราชภัฏบรจรัม

 จุฬาลงกรณ์มหาวิทยาลัย


```

for (x=0; x<NLL; ++x) {
    printf( FORM_8 , bp1[x], bq1[x], IL[0][x], IL[1][x], IL[2][x] );
    see_more_data( &k, 9 );
}
pause( "press anykey to go on" );
}

```

```

/* ----- */

```

```

PRINT_FAULT_SOLUTION()
{
    if ( ( fx = fopen( "PRN", "w" ) ) != NULL ) {
        fprintf( fx, "%c%c", 27, 77 );
        PRINT_OUTPUT();
        PRINT_BUS_CURRECT();
        PRINT_BUS_VOLTAGE();
        PRINT_LINE_CURRENT();
        fclose( fx );
    }
    else error( "Incorrect Printer Name" );
}

```

```

PRINT_OUTPUT()
{
    wait( "Printing . . . FAULT SOLUTION" );
    fprintf( fx, OUT___ );

    if ( Type == 0 ) fprintf( fx, OUT_00 , P, Phase + 65 );
    eif ( Type == 1 ) fprintf( fx, OUT_01 , P, Phase + 65 );
    eif ( Type == 2 ) fprintf( fx, OUT_02 , P, Phase + 65 );
    else fprintf( fx, OUT_03 , P, Phase + 65 );
    fprintf( fx, OUT___ );

    if ( Quan == 0 ) fprintf( fx, "\nSEQUENCE QUANTITIES\n" );
    else fprintf( fx, "\nPHASE QUANTITIES\n" );
}

```

```

PRINT_BUS_CURRECT()
{
    wait( "Printing . . . BUS CURRENT" );
    fprintf( fx, "\nBUS CURRENT\n" );

    if ( Quan == 0 ) fprintf( fx, OUT_1 );
    else fprintf( fx, OUT_1P );
    fprintf( fx, FORM_7 , P, IB[0], IB[1], IB[2] );
}

```

```

/* ----- */
/*
/*   File : MAX.B == Maximum Definition for FP1.EXE
/*
/* ----- */

#define MFLD    12      /* Max. Field for I/P */
#define SIZE    82      /* Size of each record */

#define MREC    99      /* Max. Record for I/P */
#define MRZ     55      /* Max. Record per Zone */
#define MLL     15      /* Max. Line Link */

#define MBUS    50      /* Max. BUS for I/P */
#define MBZ     30      /* Max. Bus per Zone */
#define MBL     10      /* Max. Bus Link */

#define MI      6       /* Max. Inverse Matrix */

/* ----- */

```



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```

/* ----- */
/*
/*      File : MAIN2.c
/*
/*
/*      MAIN PROGRAM for : FP1.EXE
/*
/* ----- */

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <math.h>
#define COMPLEX struct complex

#include "HAT.B"
#include "KEY.B"
#include "MAX.B"

#define eif     else if

#define D(r,f)  &data[(r)][fldaddr[(f)]]      /* addr.Data in rec,fld */
#define T(x,f)  strset( temp, 0),strncpy( temp, D((x),(f)), fldlen[(f)] )

#define CINV    fldaddr[fld]                  /* Column INVerse color */
#define CDAT    (CINV) + 1                   /* Column present Data */
#define LDAT    fldlen[fld]                  /* Length present Data */
#define LINV    (LDAT) + 2                   /* Length INVerse color */

static FILE    *fx;
static char    temp[15],                    /* temporary string */
              data[MREC][SIZE];           /* data in table */
static int     i, j,
              row = 9, rec = 0,
              top = 0, fld = 1,
              fldaddr[MFLD] = {2, 7,10,14,20,26,33,43,50,60,67,76},
              fldtype[MFLD] = {2, 2, 2, 1, 1, 3, 4, 3, 4, 3, 4, 2},
              fldint[MFLD]  = {2, 2, 2, 4, 5, 1, 2, 1, 2, 1, 2, 2},
              fldmod[MFLD]  = {0, 0, 0, 0, 0, 4, 4, 4, 4, 4, 4, 0},
              fldlen[MFLD]  = {2, 2, 2, 4, 5, 6, 7, 6, 7, 6, 7, 2};

char filename[15];                          /* data file name */
int KEY, ERR, NBUS, NINP, NLINE,           /* Keyboard
error status
Number of Buses
Number of Input rec.
Number of used Lines */
FormZone, FormZ124, ReadData, SaveData,   /* Read Data or not
Save Data or not
Form new Zbus or not */
bp[MREC], bq[MREC];                        /* Low bus No. (P)
High bus No. (Q) */
COMPLEX z[MREC];                           /* primitive impedance */

```

```
/* ----- */
```

```
main(int ac, char *av[])
```

```
{
  ReadData = ac;
  FormZbus = 1;
  SET_FAR_Z1();           /* Z1C */
  SET_FAR_ZBUS();        /* ZBUS */
  SET_VIDEO( av[1] );    /* MAUX */
  SET_MENU();            /* MENU */
  CLEAR_DATA();
  SHOW_HEAD();
  SHOW_DATA();
  TELL_CELL();
  TELL_FILENAME();
  TELL_OLDDATA();
  TELL_REQUIRE();
  TELL_DATA_CELL();
  READY_FOR_WORKING();
}
```

```
/* ----- */
```

```
CLEAR_DATA()
```

```
{
  char ch[3];
  for (i=0; i<MREC; ++i) {
    strcpy( data[i], BLANK);
    data[i][SIZE - 2] = 10;
    data[i][SIZE - 1] = 0;
  }
  for (i=0; i<9; ++i)
    data[i][3] = 49 + i;
  for (i=9; i<MREC; ++i) {
    itoa( i+1, ch, 10);
    data[i][2] = ch[0];
    data[i][3] = ch[1];
  }
  strset( filename, 0);
  strcpy( filename, "S");
  KEY = HOME;
  NINP = 0;
  NBUS = 0;
  ReadData = 1;
  FormZbus = 1;
  SaveData = 0;
}
```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



```

SHOW_HEAD()
{
    clear_screen();
    dumps( 1, 1, HAT_1), dumps( 2, 1, HAT_2), dumps( 3, 1, HAT_3);
    dumps( 5, 1, HAT_5), dumps( 6, 1, HAT_6), dumps( 7, 1, HAT_7);
    lineh( 4, 1, 80, 's'), lineh( 8, 1, 80, 's');
    linev( 5, 1, 23, 's'), linev( 5, 6, 23, 's'), linev( 5, 80, 23, 's');
    dumpc( 4, 1, 218), dumpc( 4, 6, 194), dumpc( 4, 80, 191);
    dumpc( 8, 1, 195), dumpc( 8, 6, 197), dumpc( 8, 80, 180);
    lineh(24, 1, 80, 's');
}

SHOW_DATA()
{
    dump_screen( 9, 3, 23, 4, &data[top][2], SIZE);
    dump_screen( 9, 8, 23, SIZE - 3, &data[top][7], SIZE);
}

TELL_CELL()
{
    char field, cell[4];
    itoa( rec+1, cell, 10);
    field = fld + 64;
    strcat( cell, &field, 1);
    dumps( 1, 8, " ");
    dumps( 1, 8, cell);
}

TELL_FILENAME()
{
    dumps( 1, 40, " ");
    dumps( 1, 40, filename);
}

TELL_ODDATA()
{
    clear( 2, 10, 2, 20);
    dumpsn( 2, 11, D(rec, fld), LDAT);
}

TELL_ODDATA_INV()
{
    clear( 2, 10, 2, 20);
}

```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```

color( 2, 11, REVER, LINV);
dumpsn( 2, 12, D(rec, fld), LDAT);
}

```

```
TELL_REQUIRE()
```

```

{
clear( 3, 11, 3, 60);
if (fld == 1) dumps( 3, 11, REQ_1);
eif (fld == 2) dumps( 3, 11, REQ_2);
eif (fld == 3) dumps( 3, 11, REQ_3);
eif (fld == 4) {
if (same(D(rec, 3), "GEN ", 4)) dumps( 3, 11, REQ_4a);
eif (same(D(rec, 3), "XMER", 4)) dumps( 3, 11, REQ_4b);
eif (same(D(rec, 3), "LINE", 4)) dumps( 3, 11, REQ_4c);
else
dumps( 3, 11, "I DON'T KNOW");
}
eif (fld == 5) dumps( 3, 11, REQ_5);
eif (fld == 6) dumps( 3, 11, REQ_6);
eif (fld == 7) dumps( 3, 11, REQ_7);
eif (fld == 8) dumps( 3, 11, REQ_8);
eif (fld == 9) dumps( 3, 11, REQ_9);
eif (fld == 10) dumps( 3, 11, REQ_10);
else
dumps( 3, 11, REQ_11);
}

```

```
TELL_DATA_CELL()
```

```

{
dumpsn( row, CDAT, D(rec,fld), LDAT);
color( row, CINV, REVER, LINV);
}

```

```
/* ----- */
```

```
READY_FOR_WORKING()
```

```

{
while (KEY != EXIT) {
if ( F1 < KEY && KEY < F10 ) DO_JOBS();
if ( F10 == KEY ;: KEY == '/' ) DO_MENU();
if ( 31 < KEY && KEY < 127 ) DO_INPUT();
if ( F10 < KEY && KEY < INS ) DO_MOVE();
if ( KEY == EXIT ) break;
READY_STATUS();
KEY = getkey();
}
ENDING();
}

```

ศูนย์วิทยทรัพยากร

จุฬาลงกรณ์มหาวิทยาลัย

```

READY_STATUS()
{
    dumpsa( 1, 74, " READY ", REVER);
    TELL_READY_STATUS_LINE();
    setcur( row, CDAT+LDAT/2);
}

ENDING()
{
    clear_line( 25, 25 );
    beep( 'H' );
    exit( 0 );
}

/* ----- */

DO_MENU()
{
    color(row, CINV, GREEN, LINV);
    scan_buffer( 1, 1, 25, 80);
    MENU();
    dump_buffer( 1, 1, 25, 80);
    TELL_FILENAME();
    color( row, CINV, REVER, LINV);
}

DO_MOVE()
{
    color( row, CINV, GREEN, LINV);
    if ( MOVE( KEY, MREC, MFLD, 9, 23, &row, &rec, &fld, &top)) SHOW_DATA();
    TELL_CELL();
    TELL_OLDDATA();
    TELL_REQUIRE();
    color( row, CINV, REVER, LINV);
}

DO_INPUT()
{
    dumpea( 1, 74, " INPUT ", REVER);
    TELL_OLDDATA_INV();
    if ( scan_input( temp, row, CDAT, LDAT, &KEY))
        if ( INPUT( temp, fld, fldtype, fldlen, fldint, fldmod, D( rec, 3))) {
            strncpy( D( rec, fld), temp, LDAT );
        }
}

```

```

    for (i=0; i<MREC; ++i)
        if (same( D( i, 1), " ", 5)) break;
    NINP = i;
    ReadData = 1;
    FormZone = 1;
    FormZ124 = 1;
    FormZbus = 1;
    SaveData = 1;
}
else KEY = 0;
TELL_OLDDATA();
TELL_DATA_CELL();
}

```

```
DO_JOBS()
```

```

{
    dumpsa( 1, 74, " ", GREEN);
    dumpsa( 1, 75, " JOBS ", REVER);
    switch ( KEY ) {
        case F2 : SAVE_FILE(); break;
        case F6 : PRINT_SCREEN(); break;
    }
    TELL_FILENAME();
}

```

```
/* ----- */
```

```
SAVE_FILE()
```

```

{
    if ( SaveData ) {
        if ((fx = fopen(filename, "w")) != NULL) {
            wait( "Saving . . .");
            fprintf( fx, "%d\n", NINP);
            for (i=0; i<NINP; ++i)
                fputs( data[i], fx);
            fclose( fx );
        }
        else error( "Incorrect filename");
    }
    else wait( "Saving . . . ");
}

```

```
LOAD_FILE()
```

```
{
```

ศูนย์วิทยุโทรพัชกร
จุฬาลงกรณ์มหาวิทยาลัย


```

int NR;
if (scan_subcom("Enter file name for Loading: ", temp, 8, 14, 14, &KEY)) {
    beep('H'), setcur( 1, 80);
    if ((fx = fopen( temp, "r")) != NULL) {
        fscanf( fx, "%d\n", &NR);
        if (0 < NR && NR <= MREC) {
            wait( "Loading . . .");
            CLEAR_DATA();
            NINP = NR;
            for (i=0; i<NINP; ++i)
                fgets( data[i], SIZE, fx), fscanf( fx, "%d\n", &j);
            fclose( fx );
            strcpy( filename, strupr( temp));
            ReadData = 1;
            FormZone = 1;
            FormZ124 = 1;
            FormZbus = 1;
            SaveData = 0;
            LOAD_PRE_RESULTS();
        }
        else fclose( fx ), error("Number of records in file not match.");
    }
    else error("File not found or Disk not Ready.");
}
clr_scan_subcom();
}

```

```

WRITE_FILE()
{
    if (NINP > 0) {
        if (scan_subcom("Enter new file name: ", temp, 8, 11, 14, &KEY)) {
            beep('H'), setcur( 1, 80);
            if (( fx = fopen( temp, "w")) != NULL) {
                if (confirm("Write to new File ")) {
                    wait("Saving . . .");
                    fprintf( fx, "%d\n", NINP);
                    for (i=0; i<NINP; ++i)
                        fprintf( fx, "%s", data[i]);
                    fclose( fx );
                    strcpy( filename, strupr( temp ));
                    KEY = BS;
                    SaveData = 0;
                }
                else error("Incorected File name");
            }
        }
        clr_scan_subcom();
    }
    else error("Number of input Records = 0, . . Can't save file");
}

```

```

PRINT_DATA()
{
    if ( NINP > 0 ) {
        if ( ( fx = fopen("PRN", "w") ) != NULL ) {
            wait( "Printing . . . INPUT DATA" );
            fprintf( fx, "%c%c", 27, 77 );
            fprintf( fx, "File: %s\n", filename );
            fprintf( fx, "%s", HAT_4);
            fprintf( fx, "%s", HAT_5);
            fprintf( fx, "%s", HAT_6);
            fprintf( fx, "%s", HAT_7);
            fprintf( fx, "%s", HAT_8);
            for (i=0; i<NINP; ++i)
                fprintf( fx, "%s", data[i]);
            fprintf( fx, "%s", BOTTOM);
            fprintf( fx, "\n\n");
            fclose( fx );
        }
        else error( "Printer Error" );
    }
    else error( "There are not any Data" );
}

/* ----- */

READ_DATA()
{
    wait("Read Input DATA . . . . . Loop count:"), set_count();
    NBUS = 0;
    for (j=1, i=0; i<NINP; ++i, j = i-(i/10)*10 ) {
        if ( j == 0 ) plus_count();
        T( i, 1), bp[i] = atoi( temp );
        T( i, 2), bq[i] = atoi( temp );
        T( i, 5), z[i].x = atof( temp );
        T( i, 6), z[i].y = atof( temp );
        if ( bq[i] == 0 ) break;
        NBUS = max( NBUS, max( bp[i], bq[i] ) );
        if ( NBUS > MBUS ) errors( "Line", i+1, "Too BIG Bus" ), ERR = 1;
    }
    NLINE = i;
}

/* ----- */

```

```

/* ----- */
/*
/*   File : MAUX.C
/*
/*   AUXILARY routine for : FA1.EXE & FP1.EXE
/*
/* ----- */

```

```

#include <dos.h>
#include <conio.h>
#include <string.h>
#include <stdio.h>

```

```

#define eif else if

```

```

#define REVER 120 /* Reverse green color */
#define FLASH 130 /* Blink green color */

```

```

static int VIDEO = 0xB000,
i, j, x,
off, count,
tops, lefts,
bots, rights;

```

```

static char value[7],
buf[25][80][2];

```

```

/* ----- */

```

```

SET_VIDEO(char *screen)
{
if (screen[0] == 'c') VIDEO = 0xB800;
}

```

```

/* ----- */

```

```

getkey()
{
int KEY;
KEY = bioskey(0);
if ((KEY & 0xFF) == 0) return(256 + (KEY >> 8));
else return(KEY & 0xFF);
}

```

ศูนย์วิทยทรัพยากร

จุฬาลงกรณ์มหาวิทยาลัย

```

see_more_data(int *kk, int brow)
{
    if (*kk > 22 - brow) {
        pause("Press anykey to see more data.");
        clear_line(brow, 23);
        setcur(brow, 1);    *kk = 0;
    }
    else ++*kk;
}

```

```

swapi(int *ii, int *jj)
{
    x = *ii, *ii = *jj, *jj = x;
}

```

```

pos(char ch, char *string)    /* Position of a char in String */
{
    int position = -1;
    for (j=0; string[j]!=0; ++j)
        if (ch == string[j]) { position = j; break; }
    return(position);
}

```

```

has(char ch, char *string)    /* Has a char. in String or not */
{
    int has;
    char *sear = strchr(string, ch);
    if (*sear == 0) has = 0;
    else has = 1;
    return(has);
}

```

```

have(char *str, char *string) . /* Has string in String or not */
{
    int has;
    char *sear = strstr(string, str);
    if (*sear == 0) has = 0;
    else has = 1;
    return(has);
}

```

```

same(char *s1, char *s2, int n) /* Is String.1 same String.2 */
{
    int cs, cmp = -55;
    cmp = strcmp(s1, s2, n);
    cs = 0; if (cmp == 0) cs = 1;
    return(cs);
}

```

```

/* ----- */

```

```

setcur(int row, int col)
{
    gotoxy(col, row);
}

```

```

color(int row, int col, char att, int NCHR)
{
    off = 160*row + 2*col - 161;
    for (j=0; j<NCHR; ++j, off += 2)
        pokeb( VIDEO, off, att);
}

```

```

clear(int top, int left, int bot, int right)
{
    x = top*160 + left*2 - 162;
    for (i=top; i<=bot; ++i, x+=160)
        for (off=x, j=left; j<=right; ++j, off+=2)
            pokeb( VIDEO, off, ' '), pokeb( VIDEO, off+1, GREEN);
    setcur( top, left );
}

```

```

clear_screen()
{
    clear( 1, 1, 25, 80);
}

```



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```
clear_line(int top, int bot)
{
    clear( top, 1, bot, 80);
}
```

```
/* ----- */
```

```
dumpc(int row, int col, char ch)
{
    pokeb( VIDEO, row*160 + col*2 - 162, ch);
}
```

```
dumpca(int row, int col, char ch, char att)
{
    off = row*160 + col*2 - 162;
    pokeb( VIDEO, off, ch), pokeb( VIDEO, off+1, att);
}
```

```
dumpsa(int row, int col, char *string)
{
    off = row*160 + col*2 - 162;
    for (j=0; string[j]!=0; ++j, off+=2)
        pokeb( VIDEO, off, string[j]);
}
```

```
dumpsa(int row, int col, char *string, char att)
{
    off = row*160 + col*2 - 162;
    for (j=0; string[j]!=0; ++j, off+=2)
        pokeb( VIDEO, off, string[j]), pokeb( VIDEO, off+1, att);
}
```

```
dumpsn(int row, int col, char *string, int NCHR)
{
    off = row*160 + col*2 - 162;
```

ศูนย์วิทยทรัพยากร

จุฬาลงกรณ์มหาวิทยาลัย

```

for (j=0; j<NCHR; ++j, off+=2)
    pokeb( VIDEO, off, string[j]);
}

```

```

/* ----- */

```

```

beep(char level)
{
    switch( level ) {
        case 'L': sound(250), delay(250), nosound(); break;
        case 'M': sound(500), delay(125), nosound(); break;
        case 'H': sound(1000), delay(64), nosound(); break;
    }
}

```

```

set_count()
{
    count = 0;
    itoa(count, value, 10);
    dumpsa( 25, 50, value, LIGHTRED);
}

```

```

plus_count()
{
    itoa(++count, value, 10);
    dumpsa( 25, 50, value, LIGHTRED);
}

```

```

wait(char *msg)
{
    beep('H');
    clear_line(25, 25), dumps(25, 1, msg);
    dumpsa(25, 65, "Wait a minute.", FLASH), setcur(25, 80);
}

```

```

confirm(char *msg)
{

```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```

int key;
beep('M');
clear_line( 25, 25);
dumppa( 25, 1, msg, LIGHTRED);
setcur( 25, strlen(msg) + 5), printf("Go on ? (Y/N) : ");
key = getkey();
if (key == 'Y' || key == 'y') return(1);
else return(0);
}

```

```

pause(char *msg)
{
    beep('M');
    clear_line(25, 25), dumppa(25, 1, msg);
    setcur(25, strlen(msg)+1);
    getkey();
}

```

```

/* ----- */

```

```

linev(int top, int left, int bot, char type)
{
    char line = 0xB3;
    if (type == 'd') line = 0xBA;
    for (i=top; i<=bot; ++i)
        dumppc(i, left, line);
}

```

```

lineh(int top, int left, int right, char type)
{
    char line = 0xC4;
    if (type == 'd') line = 0xCD;
    for (j=left; j<=right; ++j)
        dumppc(top, j, line);
}

```

```

frame(int top, int left, int bot, int right, char type)
{
    char t1 = 0xDA, tr = 0xBF, b1 = 0xC0, br = 0xD9;
    clear(top, left, bot, right);
}

```

ศูนย์วิทยทรัพยากร

จุฬาลงกรณ์มหาวิทยาลัย


```

if (type == 'd')  t1 = 0xC9,  tr = 0xBB,  b1 = 0xC8,  br = 0xBC;
lineh(top, left+1, right-1, type);
lineh(bot, left+1, right-1, type);
linev(top+1, left, bot-1, type);
linev(top+1, right, bot-1, type);
dumpc(top, left, t1),  dumpc(top, right, tr);
dumpc(bot, left, b1),  dumpc(bot, right, br);
}

```

```

/* ----- */

```

```

scan_buffer(int top, int left, int bot, int right)
{
  x = top*160 + left*2 - 162;
  for (i=top-1; i<bot; ++i, x+=160)
    for (off=x, j=left-1; j<right; ++j, off+=2)
      buf[i][j][0] = peekb( VIDEO, off),
      buf[i][j][1] = peekb( VIDEO, off+1);
}

```

```

dump_buffer(int top, int left, int bot, int right)
{
  x = top*160 + left*2 - 162;
  for (i=top-1; i<bot; ++i, x+=160)
    for (off=x, j=left-1; j<right; ++j, off+=2)
      pokeb( VIDEO, off, buf[i][j][0]),
      pokeb( VIDEO, off+1, buf[i][j][1]);
}

```

```

dump_screen(int top, int left, int bot, int right, char *DATA, int SIZE)
{
  int y;
  --top, --left, x = top*160 + left*2;
  for (i=top; i<bot; ++i, x+=160, DATA += SIZE)
    for (off=x, y=0, j=left; j<right; ++j, ++y, off += 2)
      pokeb( VIDEO, off, DATA[y]);
}

```

```

/* ----- */

```



```
error(char *msg)
```

```
{
    beep('L');
    scan_buffer(25, 1, 25, 80);
    clear_line(25, 25);
    dumpsa(25, 1, "!Errors!", REVER);
    dumpsa(25, 13, msg, LIGHTRED);
    dumps(25, 65, "Press any key."), setcur(25, 80);
    getkey();
    dump_buffer(25, 1, 25, 80);
}
```

```
errors(char *ms, int ii, char *msg)
```

```
{
    beep('L');
    scan_buffer(25, 1, 25, 80);
    clear_line(25, 25);
    dumpsa(25, 1, "!Errors!", REVER);
    setcur(25, 13), printf("%s %d: %s", ms, ii, msg);
    dumps(25, 65, "Press any key."), setcur(25, 80);
    getkey();
    dump_buffer(25, 1, 25, 80);
}
```

```
/*-----*/
```

```
scan_input(char *temp, int row, int col, int NCHR, int *KEY)
```

```
{
    int SCAN, colf = col - 1,
        cole = col + NCHR;
    clear(row, colf, cole);
    dumpc(row, colf, 26);
    dumpc(row, cole, 27);
    SCAN = scan_strings(temp, row, col, NCHR, KEY);
    dumpc(row, colf, 32);
    dumpc(row, cole, 32);
    return(SCAN);
}
```

```
scan_subcom(char *msg, char *temp, int row, int col, int NCHR, int *KEY)
```

```

{
    int NMSG = strlen(msg),
        colb = col + NMSG + 2;
    tops = row;
    lefts = col;
    bots = row + 2;
    rights = col + NMSG + NCHR + 4;
    scan_buffer(tops, lefts, bots, rights);
    frame(tops, lefts, bots, rights, 'd');
    dumpsa(row + 1, col + 2, msg, BROWN);
    return( scan_strings( temp, row + 1, colb, NCHR, KEY) );
}

clr_scan_subcom()
{
    dump_buffer(tops, lefts, bots, rights);
}

/* ----- */

scan_strings(char *temp, int row, int col, int NC, int *KEY)
{
    int MOVE = 1, SCAN = 0, r = row, c = col, N = 0, k = *KEY;
    if (k == 13) k = MOVE = 0;
    setcur(r, c);
    color(r, c, LIGHTRED, NC);
    TELL_SCAN_STATUS_LINE();
    while ( k != 27 ) {
        if (keychar(k))
            if ( N < NC) dumpc(r, c, k), ++c, setcur(r, c), ++N;
            else beep('L');
        if (keyback(k))
            if ( N > 0 ) --c, setcur(r, c), dumpc(r, c, ' '), --N;
            else beep('L');
        if (keymove(k))
            if ( MOVE ) { SCAN = 1; break; }
        if ( k == 13 ) { SCAN = 1; break; }
        if ( k == 320 ) PRINT_SCREEN();
        k = getkey();
    }
    if (SCAN) scans(temp, row, col, N, MOVE, KEY, k);
    else *KEY = 27;
    return( SCAN );
}

```

```

keychar(int k)
{
    if (31<k && k<127) return(1);
    else return(0);
}

keyback(int k)
{
    if (k==8 || k==127) return(1);
    else return(0);
}

keymove(int k)
{
    if (k==328 || k==331 || k==333 || k==336) return(1);
    else return(0);
}

scans(char *temp, int row, int col, int NCHR, int MOVE, int *KEY, int k)
{
    if (MOVE) *KEY = k;
    else *KEY = 0;
    for (off = row*160 + col*2 - 162, j=0; j<NCHR; ++j, off+=2)
        temp[j] = peekb( VIDEO, off);
    temp[j] = 0;
}

/* ----- ศูนย์วิจัยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย ----- */

TELL_READY_STATUS_LINE()
{
    clear_line( 25, 25);
    dumps( 25, 1, "F2-Save F6-Print F10,-MEMU");
    color( 25, 1, LIGHTCYAN, 2);
    color( 25, 12, LIGHTCYAN, 2);
    color( 25, 24, LIGHTCYAN, 3);
    color( 25, 28, LIGHTCYAN, 1);
}

```

```
TELL_SCAN_STATUS_LINE()
```

```
{
    clear_line( 25, 25);
    dumps( 25, 1, "F6-Print < -Input Esc-Go back");
    color( 25, 1, LIGHTCYAN, 2);
    color( 25, 13, LIGHTCYAN, 1);
    dumpca( 25, 14, 0xC4, LIGHTCYAN);
    dumpca( 25, 15, 0xD9, LIGHTCYAN);
    color( 25, 27, LIGHTCYAN, 3);
}
```

```
TELL_MENU_STATUS_LINE()
```

```
{
    clear_line( 25, 25);
    dumps( 25, 1, "F6-Print < -Do command Esc-Go back");
    color( 25, 1, LIGHTCYAN, 2);
    color( 25, 13, LIGHTCYAN, 1);
    dumpca( 25, 14, 0xC4, LIGHTCYAN);
    dumpca( 25, 15, 0xD9, LIGHTCYAN);
    color( 25, 32, LIGHTCYAN, 3);
}
```

```
PRINT_SCREEN()
```

```
{
    int m, n;
    unsigned char ch;
    for (m=0, i=0; i<25; ++i, m+=160) {
        for (n=0, j=0; j<80; ++j, n+=2) {
            ch = peekb( VIDEO, m+n);
            if (ch == 0 ;; ch == NULL) ch = 0x20;
            eif (ch == 0xC9 ;; ch == 0xDA) ch = 0x80;
            eif (ch == 0xBB ;; ch == 0xBF) ch = 0x81;
            eif (ch == 0xC8 ;; ch == 0xC0) ch = 0x82;
            eif (ch == 0xBC ;; ch == 0xD9) ch = 0x83;
            eif (ch == 0xBA ;; ch == 0xB3) ch = 0x84;
            eif (ch == 0xCD ;; ch == 0xC4) ch = 0x85;
            eif (ch == 0xCC ;; ch == 0xC3) ch = 0x86;
            eif (ch == 0xB9 ;; ch == 0xB4) ch = 0x87;
            eif (ch == 0xCA ;; ch == 0xC1) ch = 0x88;
            eif (ch == 0xCB ;; ch == 0xC2) ch = 0x89;
            eif (ch == 0xCE ;; ch == 0xC5) ch = 0x8A;
            write( 4, &ch, 1);
        }
        ch = 0x0A, write( 4, &ch, 1);
    }
    ch = 0x0A;
    for (i=0; i<5; ++i) write( 4, &ch, 1);
}
```

```

/* ----- */
/*                                     */
/*   File : MENU2.c                   */
/*                                     */
/*   MENU routine for : FP1.EXE      */
/*                                     */
/* ----- */

#include <bios.h>
#include <string.h>

#include "KEY.B"
#include "MAX.B"

#define MCOM      6      /* Max. COMmand      */
#define MSUB      7      /* Max. SUB-command  */

#define eif      else if

extern int      KEY,
               ERR,
               NBUS,
               FormZone,
               FormZ124,
               ReadData,
               SaveData,
               FormZbus;

static char     temp[15],
               stcom[MCOM][10], /* command string    */
               stsub[MCOM][MSUB][28]; /* sub-command string */

static int      i, j,
               docom, /* do command        */
               dosub[MCOM], /* do sub-command    */
               colcom[MCOM], cM, /* column of command */
               lencom[MCOM], lM, /* length of command */
               lensub[MCOM], lS, /* length of subcommand */
               nofsub[MCOM], nS; /* number of subcommand */

static long     time1, time2; /* Begin,End Calc.Time */

int             Type, Phase, Bus, Quan; /* parameter for calc. */
float           Time; /* Time used */
long            Memo; /* Memory used */

/* ----- */

SET_MENU()
{
    Type = 0;
    Phase = 0;

```

```

Bus = 1;
Quan = 0;
strcpy( stcom[0], "Worksheet");
strcpy( stsub[0][0], "Set Date ");
strcpy( stsub[0][1], "Set Time ");
strcpy( stsub[0][2], "Clear table");
strcpy( stcom[1], "File");
strcpy( stsub[1][0], "Load ");
strcpy( stsub[1][1], "Write to");
strcpy( stcom[2], "Calculate");
strcpy( stsub[2][0], "Type of Fault : 3 Phase ");
strcpy( stsub[2][1], "Referece Phase: A ");
strcpy( stsub[2][2], "Fault at Bus : 1 ");
strcpy( stsub[2][3], "Quantities : Sequence");
strcpy( stsub[2][4], "Fault Calculation ");
strcpy( stcom[3], "Display");
strcpy( stsub[3][0], "Zoning ");
strcpy( stsub[3][1], "Z1, Z2, Y4 ");
strcpy( stsub[3][2], "Z-BUS ");
strcpy( stsub[3][3], "Fault Solution");
strcpy( stcom[4], "Print");
strcpy( stsub[4][0], "Input Data ");
strcpy( stsub[4][1], "Zoning ");
strcpy( stsub[4][2], "Z1, Z2, Y4 ");
strcpy( stsub[4][3], "Z-BUS ");
strcpy( stsub[4][4], "Fault Solution");
strcpy( stcom[5], "Exit");
strcpy( stsub[5][0], "No ");
strcpy( stsub[5][1], "Yes");
for (i=0; i<MCOM; ++i) {
    for (j=0; stsub[i][j][0] != 0; ++j) ;
    nofsub[i] = j;
}
colcom[i] = colcom[i-1]+lencom[i-1]+6; /* column com. */
colcom[0] = 7;
lencom[0] = strlen( stcom[0]);
for (i=1; i<MCOM; ++i) {
    colcom[i] = colcom[i-1] + lencom[i-1] + 6; /* column com. */
    lencom[i] = strlen( stcom[i] ); /* length com. */
}
for (i=0; i<MCOM; ++i)
    lensub[i] = strlen( stsub[i][0] ); /* length sub. */
}

/* ----- */

DRAW_MENU()
{
    clear_line( 2, 3);
    for (i=0; i<MCOM; ++i) {
        cM = colcom[i];
    }
}

```

```

        dumpsn( 3, cM, stcom[i], lencom[i]);
        color( 3, cM, LIGHT, 1);
    }
    COLOR_MENU( 1 );
}

COLOR_MENU(int c)
{
    cM = colcom[ docom];
    lM = lencom[ docom];
    if (c == 0) color( 3, cM-1, GREEN, lM+2), color(3, cM, LIGHT, 1);
    else      color( 3, cM-1, REVER, lM+2);
}

DRAW_SUB_MENU()
{
    lS = lensub[ docom];
    nS = nofsub[ docom];
    if (nS > 0) {
        frame(4, cM-2, 5+nS, cM+lS+1, 's');
        for (i=5, j=0; j<nS; ++j, ++i)
            dumpsn( i, cM, stsub[ docom][j], lS), color( i, cM, LIGHT, 1);
    }
    COLOR_SUB_MENU(1);
}

COLOR_SUB_MENU(int c)
{
    int r = 5 + dosub[ docom];
    if ( nS > 0) {
        if ( c == 0) color( r, cM-1, GREEN, lS+2), color( r, cM, LIGHT, 1);
        else      color( r, cM-1, REVER, lS+2);
    }
}

CLEAR_SUB_MENU()
{
    dump_buffer( 4, cM-2, 5+nS, cM+lS+1);
}

COM_HOME()
{
    for (i=0; i<MCOM; ++i)
        dosub[i] = 0;
    COLOR_MENU( 0 );
}

```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย


```

CLEAR_SUB_MENU();
docom = 0;
COLOR_MENU( 1 );
DRAW_SUB_MENU();
}

```

```

COM_LEFT()
{
COLOR_MENU( 0 );
CLEAR_SUB_MENU();
if ( docom < 1) docom = MCOM-1;
else docom--;
COLOR_MENU( 1 );
DRAW_SUB_MENU();
}

```

```

COM_RIGHT()
{
COLOR_MENU( 0 );
CLEAR_SUB_MENU();
if ( docom == MCOM - 1) docom=0;
else docom++;
COLOR_MENU( 1 );
DRAW_SUB_MENU();
}

```

```

SUB_UP()
{
COLOR_SUB_MENU( 0 );
if ( dosub[ docom] < 1) dosub[ docom] = nS-1;
else dosub[ docom]--;
COLOR_SUB_MENU( 1 );
}

```

```

SUB_DOWN()
{
COLOR_SUB_MENU( 0 );
if ( dosub[ docom] == nS-1) dosub[ docom]=0;
else dosub[ docom]++;
COLOR_SUB_MENU( 1 );
}

```

```

SET_MENU_CURSOR()
{
setcur( dosub[ docom] + 5, cM);
}

```

ศูนย์วิทยทรัพยากร

จุฬาลงกรณ์มหาวิทยาลัย

```

}

/* ----- */

MENU()
{
    dumpsa( 1, 74, " ", GREEN);
    dumpsa( 1, 75, " MENU ", REVER);
    DRAW_MENU();
    DRAW_SUB_MENU();
    TELL_MENU_STATUS_LINE();
    while ( KEY != EXIT ) {
        SET_MENU_CURSOR();
        KEY = getkey();
        if ( KEY == ESC ) { KEY = BS; break; }
        eif ( KEY == HOME ) COM_HOME();
        eif ( KEY == LEFT ) COM_LEFT();
        eif ( KEY == RIGHT ) COM_RIGHT();
        eif ( KEY == UP ) SUB_UP();
        eif ( KEY == DOWN ) SUB_DOWN();
        eif ( KEY == ENTER ) { DO_COMMAND( docom, dosub[ docom]); break; }
        eif ( KEY == F6 ) PRINT_SCREEN();
    }
    GOTO_NEXT_SUB();
    if ( KEY != BS && KEY != HOME && KEY != EXIT) MENU();
}

DO_COMMAND(int com, int sub)
{
    if (com == 0)
        if (sub == 0) SET_DATE();
        eif (sub == 1) SET_TIME();
        else CLEAR_DATA();
    eif (com == 1)
        if (sub == 0) LOAD_FILE();
        else WRITE_FILE();
    eif (com == 2)
        if (sub == 0) FAULT_TYPE();
        eif (sub == 1) REFER_PHASE();
        eif (sub == 2) FAULT_AT_BUS();
        eif (sub == 3) SET_QUANTITY();
        else CALC_FAULT();
    eif (com == 3) {
        if (sub == 0 && !FormZone ) DISPLAY_ZONE();
        eif (sub == 1 && !FormZ124 ) DISPLAY_Z1_Z2_Y4();
        eif (sub == 2 && !FormZbus ) DISPLAY_ZBUS();
        eif (sub == 3 && !FormZbus ) GOTO_FAULT( 'D' );
    }
}

```

```

    else error("Re-arrange Data or Calculate before Display");
}
eif (com == 4) {
    if (sub == 0) PRINT_DATA();
    eif ( !FormZbus ) {
        if (sub == 1) PRINT_ZONE();
        eif (sub == 2) PRINT_Z1_Z2_Y4();
        eif (sub == 3) PRINT_ZBUS();
        else GOTO_FAULT( 'P' );
    }
    else error("Re-arrange Data or Calculate before Printing");
}
eif (com == 5)
    if (sub == 1) KEY = EXIT;
    else KEY = BS, beep('H');
}

GOTO_NEXT_SUB()
{
    dosub[0] = 0;
    if ( docom == 2) {
        if ( dosub[ docom] == 4) dump_buffer( 1, 1, 24, 80), dosub[ docom]=0;
        else dosub[ docom]++;
    }
    eif ( docom == 3) {
        if ( dosub[ docom] == 3) dosub[ docom]=0;
        else dosub[ docom]++;
        dump_buffer( 1, 1, 24, 80);
    }
    eif ( docom == 4) {
        if ( dosub[ docom] == 4) dosub[ docom]=0;
        else dosub[ docom]++;
    }
    dosub[5] = 0;
}

```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```

SET_DATE()
{
    if (scan_subcom( "Enter Date (MM-DD-YY): ", temp, 9, 3, 14, &KEY))
        beep('H');
    clr_scan_subcom();
}

```

```

SET_TIME()
{
    if (scan_subcom( "Enter Time (HH:MM:SS): ", temp, 9, 3, 14, &KEY))
        beep('L');
    clr_scan_subcom();
}

```

```

/* ----- */

```

```

FAULT_TYPE()
{
    clear( 14, 30, 17, 60);
    frame( 14, 30, 17, 60, 's');
    dumps( 15, 32, "1. Three Phase      (3 Phase)");
    dumps( 16, 32, "2. Line to Line      (LL)");
    if (scan_subcom("Enter Type No. : ", temp, 11, 30, 2, &KEY))
        if ( temp[0] != 0)
            if ( has( temp[0], "12")) {
                Type = temp[0] - 49;
                stsub[2][0][17] = 0;
                if ( temp[0] == '1') strcat( stsub[2][0], "3 Phase ");
                else                    strcat( stsub[2][0], "LL      ");
            }
            else error("Invalid Fault Type");
    dump_buffer( 14, 30, 17, 60);
    clr_scan_subcom();
}

```

```

REFER_PHASE()
{
    if (scan_subcom("Enter Phase ( A, B or C ) : ", temp, 11, 30, 2, &KEY))
        if ( temp[0] != 0) {
           strupr( temp );
            if (has( temp[0], "ABC")) {
                Phase = temp[0] - 65;
                stsub[2][1][17] = temp[0];
            }
            else error("Invalid Referance Phase");
        }
    clr_scan_subcom();
}

```

```

FAULT_AT_BUS()

```

```

{
if (scan_subcom("Enter Bus No. (0 for ALL) : ", temp, 11, 30, 2, &KEY))
  if ( temp[0] != 0 ) {
    Bus = atoi( temp );
    if (Bus == 0) {
      stsub[2][2][17] = 'A';
      stsub[2][2][18] = 'L';
      stsub[2][2][19] = 'L';
    }
    else {
      stsub[2][2][17] = temp[0];
      stsub[2][2][18] = temp[1];
      stsub[2][2][19] = ' ';
    }
  }
}
clr_scan_subcom();
}

```

```

SET_QUANTITY()
{
clear( 14, 30, 17, 55);
frame( 14, 30, 17, 55, 's');
dumps( 15, 32, "1. Sequence (0, 1, 2)");
dumps( 16, 32, "2. Phase (A, B, C)");
if (scan_subcom("Enter Quantity No. : ", temp, 11, 30, 2, &KEY))
  if ( temp[0] != 0 )
    if ( has( temp[0], "12") ) {
      Quan = temp[0] - 49;
      stsub[2][3][17] = 0;
      if (temp[0] == '1') strcat( stsub[2][3], "Sequence");
      else strcat( stsub[2][3], "Phase ");
    }
    else error("Invalid Quantity No.");
dump_buffer( 14, 30, 17, 55);
clr_scan_subcom();
}

```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```

LOAD_PRE_RESULTS()
{
READ_DATA();
if ( !ERR ) {
  ReadData = 0;
  LOAD_ZONE();
  if ( !ERR ) {

```

```

FormZone = 0;
LOAD_Z1();
LOAD_Z2();
LOAD_Y4();
if ( !ERR ) {
    FormZ124 = 0;
    LOAD_ZBUS();
    if ( !ERR )
        FormZbus = 0;
}
}
}
}

SAVE_PRE_RESULTS()
{
    SAVE_ZONE();
    if ( !ERR ) {
        SAVE_Z1();
        SAVE_Z2();
        SAVE_Y4();
        SAVE_ZBUS();
        if ( !ERR ) {
            FormZbus = 0;
            SaveData = 0;
        }
    }
}

CALC_FAULT()
{
    ERR = 0;
    if ( ReadData ) {
        READ_DATA();
        if ( !ERR )
            ReadData = 0;
    }
    if ( !ReadData ) {
        if ( FormZbus ) {
            FORM_ZBUS();
            if ( !ERR )
                SAVE_PRE_RESULTS();
        }
    }
    if ( !FormZbus ) GOTO_FAULT( 'D' );
}
}

```



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```

FORM_ZBUS()
(
  time1 = biostime( 0, 0 );
  SET_ZONE(); /* File: ZONE */
  if ( !ERR ) {
    FormZone = 0;
    CALC_Z1_ZC_Z2_Z4_Y4(); /* File: Z124 */
    if ( !ERR ) {
      FormZ124 = 0;
      time2 = biostime( 0, 0 );
      Time = (time2 - time1) / 18.2; /* File: ZBUS */
      CALC_ZBUS_FORM_Z1_Z2_Y4();
    }
  }
)
/* ----- */

```



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

}

MOVE_UP()

```
{
  if (*r > BR)  --*r, --*x;
  eif (*x > 0)  --*x, --*t, DUMP = 1;
}
```

MOVE_DOWN()

```
{
  if (*r < ER)  ++*r, ++*x;
  eif (*x < MX-1) ++*x, ++*t, DUMP = 1;
}
```

MOVE_LEFT()

```
{
  if (*f > 1)  --*f;
}
```

MOVE_RIGHT()

```
{
  if (*f < MF-1) ++*f;
}
```

MOVE_PAGE_UP()

```
{
  if (*t > NROW-1) *x -= NROW, *t -= NROW, DUMP = 1;
  else GO_HOME();
}
```

MOVE_PAGE_DOWN()

```
{
  if (*t < MX-NROW*2) *x += NROW, *t += NROW, DUMP = 1;
  else GO_TO_BOTTOM();
}
```



ศูนย์วิทยทรัพยากร

จุฬาลงกรณ์มหาวิทยาลัย



```
GO_TO_BOTTOM()  
{  
    *r = ER, *x = MX-1, *t = MX-NROW, *f = 1, DUMP = 1;  
}
```

```
/* ----- */
```

```
CHECK_TOP()  
{  
    int CBEG = BR - 1;  
    if (*t > 0) {  
        dumpc(CBEG, 1, 192);  
        dumpc(CBEG, 6, 193);  
        dumpc(CBEG, 80, 217);  
    }  
    else {  
        dumpc(CBEG, 1, 195);  
        dumpc(CBEG, 6, 197);  
        dumpc(CBEG, 80, 180);  
    }  
}
```

```
CHECK_BOTTOM()  
{  
    int CBOT = ER + 1;  
    if (*t == MX - NROW) {  
        dumpc(CBOT, 1, 192);  
        dumpc(CBOT, 6, 193);  
        dumpc(CBOT, 80, 217);  
    }  
    else {  
        dumpc(CBOT, 1, 196);  
        dumpc(CBOT, 6, 196);  
        dumpc(CBOT, 80, 196);  
    }  
}
```

```
/* ----- */
```



คณะแพทยศาสตร์พยาบาล
จุฬาลงกรณ์มหาวิทยาลัย

```

/* ----- */
/*
/*   File : MPUT.C
/*
/*   INPUT Routine for : FA1.EXE & FP1.EXE
/*
/* ----- */

```

```

#include <ctype.h>
#include <string.h>

```

```

#define eif     else if

```

```

static char    *D3, *TM;
static int     CANPUT,
              FD, *TY,
              *LD, *LI, *LM,
              i, j, k, ii, kk,
              mm, nn, oo, pp, rr, nmove, nfib;

```

```

/* ----- */

```

```

INPUT(char *temp, int field, int *type,
       int *lendat, int *lenint, int *lenmod, char *DA3)

```

```

{
    CANPUT = 0;
    D3 = DA3;
    TM = temp;
    FD = field;
    TY = type + FD;
    LD = lendat + FD;
    LI = lenint + FD;
    LM = lenmod + FD;
    if (strlen(TM)==1 && TM[0]==' ') { /* Blank cell */
        for (i=1; i<*LD; ++i)
            TM[i] = ' ';
        CANPUT = 1;
    }
    eif (TM[0] != ' ') { /* Input data */
        pp = strlen(TM);
        if (*TY == 1) CHECK_LABEL();
        else {
            for (mm=j=0; j<pp; ++j)
                if (isdigit(TM[j]) || TM[j]=='-' || TM[j]=='.') ++mm;
            if (mm == pp) {
                if (*TY == 2) CHECK_INTEGER();
                eif (*TY == 3) CHECK_REAL_A(); /* real + */
                eif (*TY == 4) CHECK_REAL_B(); /* real -/+ */
            }
            else error("Require only Number, Minus sign or Point.");
        }
    }
}

```

```

    }
    else error("Input by begining with blank");
    temp = TM;
    return(CANPUT);
}

```

```

/* ----- */

```

```

CHECK_LABEL()
{
   strupr(TM);
    if (FD == 3) {
        if (have(TM,"GEN XMER LINE")) CANPUT = 1;
        error("Invalid Equipment");
    }
    else if (FD == 4) {
        if (same(D3,"GEN",3)) {
            if (have(TM,"Y YG")) CANPUT = 1;
            error("Invalid generator Type");
        }
        else if (same(D3,"XMER",4)) {
            if ( (have("Y-",TM) || have("G-",TM) || have("D-",TM))
                && (have("-Y",TM) || have("-D",TM))
                && have(TM,"YG-YG YG-Y Y-YG YG-D D-YG Y-D D-Y D-D") )
                CANPUT = 1;
            else error("Invalid transformer Type");
        }
        else error("Required not thing (BLANK)");
    }
    else CANPUT = 1;
    if (CANPUT) ARRANGE_LABEL();
}

```

```

ARRANGE_LABEL()
{
    for (i=pp; i<+LD: ++i)
        TM[i] = ' ';
}

```

```

/* ----- */

```

ศูนย์วิทยทรัพยากร

จุฬาลงกรณ์มหาวิทยาลัย

```

CHECK_INTEGER()
{
    for (mm= j=0; j<pp; ++j)
        if (isdigit(TM[j])) ++mm;
    if (mm == pp) CANPUT = 1;
    else error("Require only Number");
    if (CANPUT) ARRANGE_INTEGER();
}

```

```

ARRANGE_INTEGER()
{
    if (atoi(TM) == 0) {
        for (i=0; i<*LD; ++i)
            TM[i] = ' ';
        TM[*LD-1] = '0';
    }
    else {
        nmove = *LD-pp;
        for (kk=1, k=0; k<nmove; ++k, ++kk)
            TM[*LD-kk] = TM[pp-kk], TM[pp-kk] = ' ';
        for (i=0; i<*LD; ++i)
            if (has(TM[i], "123456789")) break;
        else TM[i] = ' ';
    }
}

```

```

/*-----*/

```

```

CHECK_REAL_A()
{
    for (mm= nn= ii= oo= j=0; j<pp; ++j) {
        if (isdigit(TM[j])) ++mm;
        if (nn > 0) ++oo;
        if (TM[j] == '.') ++mm, ++nn;
        if (nn == 0) ++ii;
    }
    if (mm == pp) {
        if (ii > *LI) error("Out off range.");
        eif (oo > *LM) error("Out off preposition.");
        eif (nn > 1) error("More than One Point.");
        else CANPUT = 1;
    }
    else error("Require only Number or point.");
    if (CANPUT) {
        if (nn==1 && TM[0]!='.') SHIFT_RIGHT(), TM[0] = '0';
        eif (nn==0 && TM[0]!='.') TM[pp] = '.', ++pp;
        ARRANGE_REAL();
    }
}

```

ศูนย์วิทยากร
จุฬาลงกรณ์มหาวิทยาลัย

```

    }
}

SHIFT_RIGHT()
{
    for (kk=1, k=0; k<pp; ++k, ++kk)
        TM[pp-kk+1] = TM[pp-kk];
    ++pp;
}

/* ----- */

CHECK_REAL_B()
{
    for (mm=nn=ii=oo=rr=j=0; j<pp; ++j) {
        if (isdigit(TM[j])) ++mm;
        if ( nn > 0      ) ++oo;
        if ( TM[j] == '.' ) ++mm, ++nn;
        if ( TM[j] == '-' ) ++mm, ++rr;
        if ( nn == 0      ) ++ii;
    }
    if (MINUS_SIGN()) {
        if (mm == pp) {
            if (ii > *LI) error("Out off range.");
            eif (oo > *LM) error("Set preision in file TABLE.D again");
            eif (nn > 1)  error("More than One Point.");
            eif (nn == 0) TM[pp] = '.', ++pp, CANPUT = 1;
            else
                CANPUT=1;
        }
        else error("Input Number, Minus sign or Point.");
    }
    if (CANPUT) ARRANGE_REAL();
}

MINUS_SIGN()
{
    int sign = 0;
    if (rr > 1) error("Too many minus signs");
    eif (rr == 1) {
        if (TM[0] == '-') {
            if (pp == 1) error("Require some more");
            eif (TM[1] == '.') sign = 1, SHIFT_RIGHT(), TM[0]='-', TM[1]='0', ++mm, ++ii;
            else
                sign = 1;
        }
        else error("Misplace minus signs");
    }
}

```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



```

}
else {
    if (TM[0] == '.') {
        SHIFT_RIGHT(), TM[0]='0';
        SHIFT_RIGHT(), TM[0]=' ', mm+=2, ii+=2, sign=1;
    }
    else
        SHIFT_RIGHT(), TM[0]=' ', ++mm, ++ii, sign=1;
}
return(sign);
}

```

/* ----- */

ARRANGE_REAL()

```

{
    nfib = *LM - oo;
    for (k=0; k<nfib; ++k)
        TM[pp+k] = '0';
    pc += k;
    nmove = *LD - pp;
    for (kk=1, k=0; k<nmove; ++k, ++kk)
        TM[*LD-kk] = TM[pp-kk], TM[pp-kk] = ' ';
}

```

/* ----- */

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```

/* ----- */
/*
/*   File : COMPLEX.c
/*
/*   COMPLEX founction for : FA1.EXE & FP1.EXE
/*
/* ----- */

```

```

#include <stdio.h>
#include <math.h>
#define COMPLEX struct complex

```

```

#define eif    else if

```

```

#define P      M_PI_2
#define PI     M_PI
#define PI2    M_PI*2

```

```

#define min    1E-4
#define MIN    1E-100
#define MAX    1E+3

```

```

/* ----- */

```

```

cadd( Z, zF, zB )      COMPLEX *Z, zF, zB;      /* Complex add */
{
    Z->x = zF.x + zB.x;
    Z->y = zF.y + zB.y;
    modify_rectan( Z );
}

```

```

cstt( Z, zF, zB )     COMPLEX *Z, zF, zB;      /* Com.subtract */
{
    Z->x = zF.x - zB.x;
    Z->y = zF.y - zB.y;
    modify_rectan( Z );
}

```

```

cmul( Z, zF, zB )     COMPLEX *Z, zF, zB;      /* Com.multiple */
{
    polar( &zF );
    polar( &zB );
    Z->x = zF.x * zB.x;
    Z->y = zF.y + zB.y;
}

```

```

modify_polar( Z );
rectan( Z );
}

```

```

cdiv( Z, zF, zB )      COMPLEX *Z, zF, zB;      /* Comp. divide */
{
    polar( &zF );
    polar( &zB );
    if ( zB.x > MIN ) Z->x = zF.x / zB.x;
    else              Z->x = MAX , printf("\nDivide by 0. ");
    Z->y = zF.y - zB.y;
    modify_polar( Z );
    rectan( Z );
}

```

```

/* ----- */

```

```

polar( COMPLEX *Z )      /* rectan to Polar in Radius */
{
    COMPLEX T = *Z;
    Z->x = cabs( T );
    if ( T.x < -MIN ) Z->y = PI + atan( T.y / T.x );
    eif ( T.x > MIN ) Z->y = atan( T.y / T.x );
    else
        if ( T.y < -MIN ) Z->y = -P;
        eif ( T.y > MIN ) Z->y = P;
        else Z->y = 0;
    if ( Z->x < MIN ) Z->x = 0;
}

```

```

modify_polar( COMPLEX *Z )
{
    if ( Z->x < MIN ) Z->x = 0;
    if ( Z->y > PI2 ) Z->y -= PI2;
    eif ( Z->y < -PI2 ) Z->y += PI2;
}

```

```

rectan( COMPLEX *Z )
{
    COMPLEX U = *Z;

```

ศูนย์วิทยทรัพยากร

จุฬาลงกรณ์มหาวิทยาลัย


```

Z->x = U.x * cos( U.y );
Z->y = U.x * sin( U.y );
modify_rectan( Z );
}

```

```

modify_rectan( COMPLEX *Z )
{
  if ( fabs( Z->x ) < MIN ) Z->x = 0;
  if ( fabs( Z->y ) < MIN ) Z->y = 0;
}

```

```

/* ----- */

```

```

polar( COMPLEX *Z )          /* rectan to Polar in Degree */
{
  polar( Z );
  if ( Z->x < min )          Z->y = 0;
  else {                    Z->y *= 180/PI;
    if ( Z->y > 180 )        Z->y -= 360;
    eif ( Z->y < -180 )     Z->y += 360;
  }
}

```

```

/* ----- */

```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```

/* ----- */
/*
/*      File :  N1_ZONE.c
/*
/*      ZONING routine for :  FP1.EXE
/*
/* ----- */

#include <stdio.h>
#include <string.h>

#include "MAX.B"
#include "FORM.B"

#define eif      else if

extern int      ERR,
               NBUS,
               NLINE,
               bp[MREC],
               bq[MREC];

extern char    filename[15];

static int     i, j, r,
               D, NBA, NBB,
               NUSE, NCHK,
               use[MREC],
               Memo[MBZ],
               PinA, PinB,
               QinA, QinB,
               NGEN, LONG,
               DEC[2], MEM_USED;

static char    P, Q, L,
               cutline[MREC+1],
               GenBus[MBZ+1],
               RemBus[MBUS+1],
               MovBus[MBZ+1],
               rebus[2][MBZ+1],
               buszone[2][MBZ+1];

static FILE    *fx;

char           BusZone[2][MBZ+1],      /* Buses in Zone A, B */
               CutLine[MI+1],          /* Cut-Line */
               LineZone[2][MRZ+1];     /* Lines in Zone A, B */

int           NB[2], NLC;              /* Number of buses,line */

/* ----- */

```

```

SET_ZONE()
{
    wait( "Calculating . . . Zoning      Loop count:" ), set_count();
    NGEN = NUMBER_OF_GENERATOR_BUSES();
    if ( NGEN == 0 ) error( "System hasn't Generator." ), ERR = 1;
    eif ( NGEN == 1 ) SET_ONLY_ZONE_A();
    eif ( NGEN <= MBZ ) SET_TWO_ZONE();
    else
        error( "System has too many GENs" ), ERR = 1;
    if ( !ERR ) {
        NB[A] = strlen( BusZone[A] );
        NB[B] = strlen( BusZone[B] );
        NLC = strlen( CutLine );
        if ( NB[A] > MBZ ) error( "Too many Buses in zone A" ), ERR = 1;
        if ( NB[B] > MBZ ) error( "Too many Buses in zone B" ), ERR = 1;
        if ( NLC > MI ) error( "Too many Cutlines . . ." ), ERR = 1;
    }
    if ( !ERR ) SET_ORDER();
}

```

```

NUMBER_OF_GENERATOR_BUSES()
{
    strset( GenBus, 0 );
    for ( i=0; i<NLINE; ++i )
        if ( bp[i] == 0 )
            if ( !has( bq[i], GenBus ) )
                Q = bq[i], strncat( GenBus, &Q, 1);
    return( strlen( GenBus ) );
}

```

/*-----ศูนย์วิทยพัทยากร-----*/

```

SET_ONLY_ZONE_A()
{
    strset( BusZone[A], 0 );
    for ( i=0; i<NBUS; ++i ) {
        Q = i+1;
        strncat( BusZone[A], &Q, 1);
    }
    strset( BusZone[B], 0 );
    strset( CutLine, 0 );
}

```

ศูนย์วิทยพัทยากร
จุฬาลงกรณ์มหาวิทยาลัย

```

SET_TWO_ZONE()
{
    BEGINNING_BUS_FOR_EACH_ZONE();
    FIRST_PARAMETER();
    SECOND_PARAMETER();
    COLLECT_BUSES_FOR_EACH_ZONE();
}

/* ----- */

BEGINNING_BUS_FOR_EACH_ZONE()
{
    int x, y, a, b, DIST;
    LONG = 0;
    for (x=0; x<NGEN; ++x)
        for (y=x+1; y<NGEN; ++y) {
            plus_count();
            DIST = DISTANCE_AB( x, y);
            if ( DIST > LONG )
                LONG = DIST , a = x, b = y;
        }
    if ( GenBus[a] > GenBus[b] ) {
        r = GenBus[a], GenBus[a] = GenBus[b], GenBus[b] = r;
    }
    strset( BusZone[A], 0 ), strncpy( BusZone[A], &GenBus[a], 1);
    strset( BusZone[B], 0 ), strncpy( BusZone[B], &GenBus[b], 1);
}

DISTANCE_AB(int aa, int bb)
{
    int G = GenBus[bb], STOP = 0, DISTANCE = 0;
    strset( buszone[A], 0 );
    strncpy( buszone[A], &GenBus[aa], 1);
    for (i=0; i<NLINE; ++i) use[i] = 0;
    while ( !STOP ) {
        ++DISTANCE;
        strset( RemBus, 0 );
        for (i=0; i<NLINE; ++i)
            if ( !use[i] ) {
                P = bp[i], Q = bq[i];
                PinA = has( P, buszone[A]);
                QinA = has( Q, buszone[A]);
                if ( P==0 ) use[i] = 1;
                eif ( PinA ) {
                    if ( Q==G ) { STOP = 1; break; }
                }
            }
    }
}

```

ศูนย์วิทยทรัพยากร
มหาวิทยาลัย

```

        eif ( !QinA )   strncat( RemBus, &Q, 1), use[i] = 1;
    }
    eif ( QinA )   {
        if ( P==G )   { STOP = 1; break; }
        eif ( !PinA )   strncat( RemBus, &P, 1), use[i] = 1;
    }
}
strcat( buszone[A], RemBus);
}
return( DISTANCE );
}

```

```

/* ----- */

```

```

FIRST_PARAMETER()
{
    int NLOOP = (LONG-2)/2;
    strset( buszone[A], 0 ); strcpy( buszone[A], BusZone[A]);
    strset( buszone[B], 0 ); strcpy( buszone[B], BusZone[B]);
    COLLECT_SIDE_BUS_ZONE( A, NLOOP );
    COLLECT_SIDE_BUS_ZONE( B, NLOOP );
    SHIFT_LEFT_buszone( A );
    SHIFT_LEFT_buszone( B );
    strcat( BusZone[A], buszone[A]);
    strcat( BusZone[B], buszone[B]);
}

```

```

COLLECT_SIDE_BUS_ZONE(int D, int DIST)
{
    for (i=0; i<NLINE; ++i) use[i] = 0;
    for (i=0; i<DIST; ++i) {
        strset( RemBus, 0 );
        for (r=0; r<NLINE; ++r)
            if ( !use[r] ) {
                P = bp[r], Q = bq[r];
                PinA = has( P, buszone[D]);
                QinA = has( Q, buszone[D]);
                if ( P==0 ) use[r] = 1;
                eif ( PinA ) {
                    if ( QinA ) use[r] = 1;
                    eif ( !QinA ) strncat( RemBus, &Q, 1), use[r] = 1;
                }
                eif ( QinA ) {
                    if ( PinA ) use[r] = 1;
                    eif ( !PinA ) strncat( RemBus, &P, 1), use[r] = 1;
                }
            }
    }
}

```

ศูนย์วิจัยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```

    }
    strcat( buszone[D], RemBus);
}
}

```

```
SHIFT_LEFT_buszone(int D)
```

```

{
    int NB = strlen( buszone[D] );
    for (i=0; i<NB; ++i) {
        P = buszone[D][i];
        for (j=i+1; j<NB; ++j) {
            if ( P == buszone[D][j] ) {
                strcpy( &buszone[D][j], &buszone[D][j+1] );
                --j, --NB;
            }
        }
    }
    strcpy( buszone[D], &buszone[D][1] );
}
}

```

```
/* ----- */
```

```
SECOND_PARAMETER()
```

```

{
    int NGZ;
    NBA = strlen( BusZone[A] );
    NBB = strlen( BusZone[B] );
    for (i=0; i<NLINE; ++i) use[i] = 0;
    for (r=0; r<NBA; ++r) use[ BusZone[A][r] ] = 1;
    for (r=0; r<NBB; ++r) use[ BusZone[B][r] ] = 1;
    strset( RemBus, 0 );
    for (Q=1, i=0; i<NBUS; ++i, ++Q) {
        if ( !use[Q] )
            strcat( RemBus, &Q, 1);
    }
    NGZ = NBA + NBB;
    NBA = (NBUS - NGZ + 1) / 2;
    strset( rebus[A], 0 ); strcpy( rebus[A], RemBus, NBA );
    strset( rebus[B], 0 ); strcpy( rebus[B], &RemBus[NBA] );
    SET_buszone( A );
    SET_buszone( B );
    MEM_USED = CALC_MEMORY();
}

```

```
/* ----- */
```

```
COLLECT_BUSES_FOR_EACH_ZONE()
```

```
{
    int DECREASE, STOP = 0;
    DEC[A] = DEC[B] = 0;
    while ( !STOP ) {
        DECREASE = 0;
        if ( MOVE_BUS( A, B ) ) ++DECREASE;
        if ( MOVE_BUS( B, A ) ) ++DECREASE;
        if ( DECREASE == 0 ) STOP = 1;
    }
    strcat( BusZone[A], rebus[A] );
    strcat( BusZone[B], rebus[B] );
    SET_CUTLINE();
}
```

```
MOVE_BUS(int X, int Y)
```

```
{
    int x, Pt, MEM_LOOP = MEM_USED, NREM = strlen( rebus[X] );
    plus_count();
    DEC[X] = 0;
    if ( DEC[Y] == 1 ) --NREM;
    for ( x=0; x<NREM; ++x ) {
        MovBus[x] = rebus[X][0];
        strcpy( rebus[X], &rebus[X][1] );
        SET_buszone( X );
        SET_buszone( Y );
        strcat( buszone[Y], &MovBus[x], 1);
        Memo[x] = CALC_MEMORY();
        strcat( rebus[X], &MovBus[x], 1);
    }
    for ( x=0; x<NREM; ++x ) {
        if ( Memo[x] < MEM_LOOP )
            MEM_LOOP = Memo[x], Pt = x;
    }
    if ( MEM_LOOP < MEM_USED ) {
        MEM_USED = MEM_LOOP;
        strcat( rebus[Y], &MovBus[Pt], 1);
        if ( DEC[Y] == 1 ) ++Pt;
        strcpy( &rebus[X][Pt], &rebus[X][Pt+1] );
        DEC[X] = 1;
    }
    return( DEC[X] );
}
```

ศูนย์วิทยุพยากรณ์
จุฬาลงกรณ์มหาวิทยาลัย

```

SET_buszone(int D)
{
    strset( buszone[D], 0 );
    strcpy( buszone[D], BusZone[D] );
    strcat( buszone[D], rebus[D] );
}

CALC_MEMORY()
{
    int MEMORY;
    strset( cutline, 0 );
    for (L=1, i=0; i<NLINE; ++i, ++L) {
        PinA = has( bp[i], buszone[A] );
        PinB = has( bp[i], buszone[B] );
        QinA = has( bq[i], buszone[A] );
        QinB = has( bq[i], buszone[B] );
        if ((PinA && QinB) || (PinB && QinA)) strncat( cutline, &L, 1);
    }
    NBA = strlen( buszone[A] );
    NBB = strlen( buszone[B] );
    NLC = strlen( cutline );
    MEMORY = NBA*NBA + NBB*NBB + NLC*NLC;
    return( MEMORY );
}

SET_CUTLINE()
{
    strset( CutLine, 0 );
    for (L=1, i=0; i<NLINE; ++i, ++L) {
        PinA = has( bp[i], BusZone[A] );
        PinB = has( bp[i], BusZone[B] );
        QinA = has( bq[i], BusZone[A] );
        QinB = has( bq[i], BusZone[B] );
        if ((PinA && QinB) || (PinB && QinA)) strncat( CutLine, &L, 1);
    }
}

```

/* ----- */


```

SET_ORDER()
{
    D = A;
    CALC_NLIN();
    ORDER_GEN();
    ORDER_LINE();
    D = B;
    CALC_NLIN();
    ORDER_GEN();
    ORDER_LINE();
}

```

```

CALC_NLIN()
{
    NUSE = 0;
    for (i=0; i<NLINE; ++i) {
        if ( bp[i] == 0 ) {
            if ( has( bq[i], BusZone[D] )) use[i] = 0;
            else ++NUSE, use[i] = 1;
        }
        elif ( has( bp[i], BusZone[D] )) {
            if ( has( bq[i], BusZone[D] )) use[i] = 0;
            else ++NUSE, use[i] = 1;
        }
        else ++NUSE, use[i] = 1;
    }
}

```

```

ORDER_GEN()
{
    memset( BusZone[D], 0 );
    memset( LineZone[D], 0 );
    for (i=0; i<NLINE; ++i)
        if ( !use[i] )
            Collect_Data( i, bq[i] );
}

```

```

Collect_Data( int rr, char Q )
{
    char L = rr + 1;
    strncat( LineZone[D], &L, 1);
    if ( !has( Q, BusZone[D] )) strncat( BusZone[D], &Q, 1);
    use[rr] = 1;
    ++NUSE;
}

```



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```

ORDER_LINE()
{
    while ( NUSE < NLINE ) {
        NCHK = NUSE;
        for (i=0; i<NLINE; ++i)
            if ( !use[i] )
                if (has( bp[i], BusZone[D]))
                    Collect_Data( i, bq[i] ), j = i + 1;
        if (NCHK == NUSE)
            for (i=0; i<NLINE; ++i)
                if ( !use[i] )
                    if ( has( bq[i], BusZone[D] ) )
                        { swapi( &bp[i], &bq[i] );
                          Collect_Data( i, bq[i] ); break; }
        if (NCHK == NUSE)
            { errors( "Line", j, "Can't go on to Ordering" ), ERR = 1; break; }
    }
}

/* ----- */

#define ZONES "-----\nLIST OF ZONES\n-----"

DISPLAY_ZONE()
{
    clear_line( 1, 23);
    printf( ZONES );
    printf( "\n\n\nBUSES IN ZONE A : " );
    for (j=1, i=0; i<NB[A]; ++i, j = i-(i/10)*10 ) {
        if (j == 0) printf( "\n\n\n" );
        printf( " %3d ", BusZone[A][i]);
    }
    printf( "\n\n\nBUSES IN ZONE B : " );
    for (j=1, i=0; i<NB[B]; ++i, j = i-(i/10)*10 ) {
        if (j == 0) printf( "\n\n\n" );
        printf( " %3d ", BusZone[B][i]);
    }
    printf( "\n\n\nLIST OF TIE LINE : " );
    for (j=1, i=0; i<NLC; ++i, j = i-(i/5)*5 ) {
        r = CutLine[i] - 1;
        if (j == 0) printf( "\n\n\n" );
        printf( " %3d -%3d ", bp[r], bq[r] );
    }
}

```

```

pause( "Press any key to go back." );
}

PRINT_ZONE()
{
    if (( fx = fopen( "PRN", "w" )) != NULL ) {
        wait( "Printing . . .          LIST OF ZONES" );
        fprintf( fx, "%c%c", 27, 77);
        fprintf( fx, ZONES);
        fprintf( fx, "\n\n\nBUSES IN ZONE  A :  " );
        for (j=1, i=0; i<NB[A]; ++i, j = i-(i/10 )*10 ) {
            if (j == 0 ) fprintf( fx, "\n\n          " );
            fprintf( fx, " %3d ", BusZone[A][i]);
        }
        fprintf( fx, "\n\n\nBUSES IN ZONE  B :  " );
        for (j=1, i=0; i<NB[B]; ++i, j = i-(i/10 )*10 ) {
            if (j == 0 ) fprintf( fx, "\n\n          " );
            fprintf( fx, " %3d ", BusZone[B][i] );
        }
        fprintf( fx, "\n\n\nLIST OF TIE LINE :  " );
        for (j=1, i=0; i<NLC; ++i, j = i-(i/5)*5 ) {
            r = CutLine[i] - 1;
            if (j == 0 ) fprintf( fx, "\n\n          " );
            fprintf( fx, " %3d -%3d  ", bp[r], bq[r] );
        }
        fprintf( fx, "\n\n" );
        fclose( fx );
    }
    else error( "Incorrect Printer Name." );
}

```

```

SAVE_ZONE()
{
    int BB, LL;
    char fileZ[15];
    strcpy( fileZ, filename );
    strcat( fileZ, "D" );
    if (( fx = fopen( fileZ, "w" )) != NULL ) {
        wait( "Saving ZONE file . . ." );
        for (D=0; D<2; ++D) {
            fprintf( fx, "%d\n", NB[D] );
            for (i=0; i<NB[D]; ++i)
                BB = BusZone[D][i], fprintf( fx, "%3d", BB );
            fprintf( fx, "\n" );
        }
        fprintf( fx, "%d\n", NLC );
        for (i=0; i<NLC; ++i)
            LL = CutLine[i], fprintf( fx, "%3d", LL );
    }
}

```

ศูนย์วิทยทรัพยากร
มหาวิทยาลัย

```

    fprintf( fx, "\n" );
    fclose( fx );
}
else error( "Disk Drive not ready." ), ERR = 1;
}

```

```

LOAD_ZONE()
{
    int BB, LL;
    char fileZ[15];
    strcpy( fileZ, filename );
    strcat( fileZ, "D" );
    if ( ( fx = fopen( fileZ, "r" ) ) != NULL ) {
        wait( "Loading ZONE file . . ." );
        for ( D=0; D<2; ++D ) {
            fscanf( fx, "%d\n", &NB[D] );
            for ( i=0; i<NB[D]; ++i )
                fscanf( fx, "%d", &BB ), BusZone[D][i] = BB;
            fscanf( fx, "\n" );
            BusZone[D][i] = 0;
        }
        fscanf( fx, "%d\n", &NLC );
        for ( i=0; i<NLC; ++i )
            fscanf( fx, "%d", &LL ), CutLine[i] = LL;
        fscanf( fx, "\n" );
        CutLine[i] = 0;
        fclose( fx );
    }
    else ERR = 1;
}

```



/* ----- * /

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```

/* ----- */
/*
/*   File : N2_Z124.c
/*
/*       Z1, Z2, Y4 Calculate routine for : FP1.EXE
/*
/* ----- */

```

```

#include <stdio.h>
#include <alloc.h>
#include <stdlib.h>
#include <string.h>
#include <math.h>
#define COMPLX struct complex

#define eif else if

#include "MAX.B"
#include "FORM.B"

extern char filename[15],
LineZone[2][MRZ+1],
BusZone[2][MBZ+1],
CutLine[MI+1];

extern int NB[2], NLC,
ERR,
NBUS,
bp[MREC],
bq[MREC];

extern COMPLX z[MREC];

extern float Time;
extern long Memo;

static int i, j, k, X, Y,
D, P, Q, L, /* Zone, bus P,Q,L */
PA, PB, QA, QB, /* count meet bus Q */
Cbus[MBZ+1];

static char fileZ[15];

static double Vx, Vy;

static COMPLX Znear, Ztemp,
C10 = {1,0}, /* Z.x = 1, Z.y = 0 */
C00 = {0,0}, /* Z.x = 0, Z.y = 0 */
Zpqpq, Zbuf, /* Z in step, Z buffer */
ZI[MI][MI]; /* Inv. matrix */

static FILE *fx;

```

```

COMPLX far *Z1[2][MBZ+1]; /* Z1 matrix */

COMPLX ZC[MI][MI], /* ZCutline */
Z2[2][MBZ][MI], /* Z2 matrix */
Z4[MI][MI], /* Z4 matrix */
Y4[MI][MI]; /* Y4 matrix */

/* ----- */

SET_FAR_Z1()
{
    long size1 = 16 * (MBZ+1);
    for (i=0; i<2; ++i) {
        for (j=0; j<=MBZ; ++j) {
            Z1[i][j] = farmalloc( size1 );
            if ( Z1[i][j] == NULL ) error( "Not enough room for Data" ), exit(1);
        }
    }
    for (i=0; i<2; ++i)
        for (j=0; j<=MBZ; ++j)
            for (k=0; k<=MBZ; ++k)
                Z1[i][j][k].x = Z1[i][j][k].y = 0;
}

/* ----- */

CALC_Z1_ZC_Z2_Z4_Y4()
{
    wait( "Calculating . . . Z1 zone A Loop count:" ), set_count();
    D = A, CALC_Z1();
    wait( "Calculating . . . Z1 zone B Loop count:" ), set_count();
    D = B, CALC_Z1();
    wait( "Calculating . . . Z Cutline Loop count:" ), set_count();
    CALC_ZC();
    wait( "Calculating . . . Z2 Loop count:" ), set_count();
    CALC_Z2();
    wait( "Calculating . . . Z4 Loop count:" ), set_count();
    CALC_Z4();
    wait( "Calculating . . . Y4 Loop count:" ), set_count();
    CALC_Y4();
    CALC_MEMO();
}

```

```

CALC_Z1()
{
    int x, line, rec, bP, bQ, H = 0,
        NL = strlen( LineZone[D] ),
        NB = strlen( BusZone[D] );
    for (i=0; i<NB; ++i)
        Cbus[i] = 0;
    for (x=0; x<NL; ++x) {
        plus_count();
        line = LineZone[D][x]; /* line used */
        rec = line - 1; /* record used */
        bP = bp[rec]; /* bus P */
        bQ = bq[rec]; /* bus Q */
        P = pos( bP, BusZone[D] ); /* position P */
        Q = pos( bQ, BusZone[D] ); /* position Q */
        H = max(max( H, P), max( H, Q)); /* High bus */
        L = H + 1; /* position L */
        Zppq = z[rec]; /* Z line */
        CREATE_ZBUS_BY(SET_FORMULA( bP ));
    }
}

```

```

SET_FORMULA(int BP)
{
    int Formula;
    if (++Cbus[Q] < 2)
        if (BP == 0) Formula = 1;
        else Formula = 2;
    else
        if (BP == 0) Formula = 3;
        else Formula = 4;
    return(Formula);
}

```

```

CREATE_ZBUS_BY(int FORMULA)
{
    switch ( FORMULA ) {
        case 1: CALC_FORMULA_1_(); break;
        case 2: CALC_FORMULA_2_(); break;
        case 3: CALC_FORMULA_3_(); break;
        case 4: CALC_FORMULA_4_(); break;
    }
}

```

ศูนย์วิทยทรัพยากร

จุฬาลงกรณ์มหาวิทยาลัย

```

CALC_FORMULA_1_( )
{
  for (i=0; i<L; ++i) {
    plus_count();
    if (i != Q)
      Z1[D][Q][i] = Z1[D][i][Q] = C00;
  }
  Z1[D][Q][Q] = Zpqpq;
}

```

```

CALC_FORMULA_2_( )
{
  for (i=0; i<L; ++i) {
    plus_count();
    if (i != Q)
      Z1[D][Q][i] = Z1[D][i][Q] = Z1[D][P][i];
  }
  cadd( &Znear, Z1[D][P][Q], Zpqpq ), Z1[D][Q][Q] = Znear;
}

```

```

CALC_FORMULA_3_( )
{
  for (i=0; i<L; ++i) {
    plus_count();
    cstt( &Znear, C00, Z1[D][Q][i] );
    Z1[D][L][i] = Z1[D][i][L] = Znear;
  }
  cstt( &Znear, Zpqpq, Z1[D][Q][L] ), Z1[D][L][L] = Znear;
  ELIMINATE_BUS_L_( );
}

```

```

CALC_FORMULA_4_( )
{
  for (i=0; i<L; ++i) {
    plus_count();
    cstt( &Znear, Z1[D][P][i], Z1[D][Q][i] );
    Z1[D][L][i] = Z1[D][i][L] = Znear;
  }
  cstt( &Zbuf, Z1[D][P][L], Z1[D][Q][L] );
  cadd( &Znear, Zbuf, Zpqpq ), Z1[D][L][L] = Znear;
  ELIMINATE_BUS_L_( );
}

```

ศูนย์วิทยทรัพยากร

จุฬาลงกรณ์มหาวิทยาลัย



```
ELIMINATE_BUS_L_()
{
  for (i=0; i<L; ++i) {
    for (j=0; j<L; ++j) {
      plus_count();
      cdiv( &Zbuf, Z1[D][i][L], Z1[D][L][L] );
      cmul( &Zbuf, Zbuf, Z1[D][L][j] );
      cstt( &Znear, Z1[D][i][j], Zbuf ), Z1[D][i][j] = Znear;
    }
  }
}
```

```
/*-----*/
```

```
CALC_ZC()
{
  for (i=0; i<NLC; ++i) {
    plus_count();
    for (j=0; j<NLC; ++j)
      if ( i == j ) ZC[i][j] = z[ CutLine[i] - 1 ];
      else ZC[i][j] = COO;
  }
}
```

```
/*-----*/
```

ศูนย์วิทยทรัพยากร จุฬาลงกรณ์มหาวิทยาลัย

```
CALC_Z2()
{
  for (j=0; j<NLC; ++j) {
    plus_count();
    P = bp[ CutLine[j] - 1 ];
    Q = bq[ CutLine[j] - 1 ];
    PA = pos( P, BusZone[A] );
    PB = pos( P, BusZone[B] );
    QA = pos( Q, BusZone[A] );
    QB = pos( Q, BusZone[B] );
    if ( PA > -1 && QB > -1 ) {
      for (i=0; i<NB[A]; ++i) cstt( &Z2[A][i][j], COO, Z1[A][i][PA] );
    }
  }
}
```

```

        for (i=0; i<NB[B]; ++i)          Z2[B][i][j] = Z1[B][i][QB] ;
    }
    eif ( PB > -1 && QA > -1 ) --{
        for (i=0; i<NB[A]; ++i)          Z2[A][i][j] = Z1[A][i][QA] ;
        for (i=0; i<NB[B]; ++i)          cstt( &Z2[B][i][j], COO, Z1[B][i][PB] );
    }
    else
        errors( "CutLine: ", j, "Incorrect in Z2." ), ERR = 1;
}
}

```

```
/* ----- */
```

```

CALC_Z4()
{
    plus_count();
    for (i=0; i<NLC; ++i)
        for (j=0; j<NLC; ++j)
            Z4[i][j] = ZC[i][j];
    for (i=0; i<NLC; ++i) {
        plus_count();
        P = bp[ CutLine[i] - 1 ];
        Q = bq[ CutLine[i] - 1 ];
        PA = pos( P, BusZone[A] );
        PB = pos( P, BusZone[B] );
        QA = pos( Q, BusZone[A] );
        QB = pos( Q, BusZone[B] );
        if ( PA > -1 && QB > -1 ) {
            for (j=0; j<NLC; ++j)
                cstt( &Z4[i][j], Z4[i][j], Z2[A][PA][j] ),
                cadd( &Z4[i][j], Z4[i][j], Z2[B][QB][j] );
        }
        eif ( PB > -1 && QA > -1 ) {
            for (j=0; j<NLC; ++j)
                cadd( &Z4[i][j], Z4[i][j], Z2[A][QA][j] ),
                cstt( &Z4[i][j], Z4[i][j], Z2[B][PB][j] );
        }
    }
    else
        errors( "CutLine: ", j, "Incorrect in Z4." ), ERR = 1;
}
}

```

```
/* ----- */
```

```
CALC_Y4()
```

```
{
    plus_count();
    for (i=0; i<NLC; ++i)
        for (j=0; j<NLC; ++j)
            ZI[i][j] = Z4[i][j];
    plus_count();
    INVERSE_ZI(NLC);
    if (!ERR) {
        plus_count();
        for (i=0; i<NLC; ++i)
            for (j=0; j<NLC; ++j)
                Y4[i][j] = ZI[i][j];
    }
}
```

```
INVERSE_ZI(int DIM)
```

```
{
    int I, II;
    double AMAX;
    COMPLEX ZP[M][MI*2];
    for (i=0; i<DIM; ++i) { /* [ZI ; Unit_matrix ] */
        for (j=0; j<DIM; ++j) {
            ZP[i][j] = ZI[i][j];
            if (j == i) ZP[i][j+DIM] = C10;
            else ZP[i][j+DIM] = C00;
        }
    }
    for (i=0; i<DIM; ++i) {
        plus_count();
        if (cabs(ZP[i][i]) < 1.0E-10)
            if (i == DIM-1) { ERR = 1; break; }
        AMAX = cabs(ZP[i][i]), I = i;
        for (II=i+1; II<DIM; ++II)
            if (cabs(ZP[II][II]) > AMAX)
                AMAX = cabs(ZP[II][II]), I = II;
        plus_count();
        if (AMAX < 1.0E-10) { ERR = 1; break; }
        for (j=0; j<DIM*2; ++j)
            Ztemp = ZP[I][j], ZP[I][j] = ZP[i][j], ZP[i][j] = Ztemp;
        Zbuf = ZP[i][i];
        plus_count();
        for (j=0; j<DIM*2; ++j)
            cdiv( &ZP[i][j], ZP[i][j], Zbuf);
        plus_count();
        for (k=0; k<DIM; ++k) {
            if (k != i) {
                Zbuf = ZP[k][i];
                for (j=0; j<DIM*2; ++j) {
```

มหาวิทยาลัยเทคโนโลยีพระยากร
 คณะวิศวกรรมศาสตร์
 มหาวิทยาลัยเทคโนโลยีพระยากร

```

        cmul( &Ztemp, Zbuf, ZP[i][j] );
        cstt( &ZP[k][j], ZP[k][j], Ztemp);
    }
}
}
}
if (IERR) {
    plus_count();
    for (i=0; i<DIM; ++i) /* [ZI] <- [ZP] */
        for (j=0; j<DIM; ++j)
            ZI[i][j] = ZP[i][j+DIM];
}
else error( "This Matrix can not Inverse" );
}

```

```

/* ----- */

```

```

CALC_MEMO()

```

```

{
    int NZ1A = NB[A] * NB[A],
        NZ1B = NB[B] * NB[B],
        NZC = NLC * NLC;
    Memo = NZ1A + NZ1B + NZC;
}

```

```

/* ----- */

```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```

DISPLAY_Z1_Z2_Y4()

```

```

{
    DISPLAY_Z1();
    DISPLAY_Z2();
    DISPLAY_Y4();
}

```

```

DISPLAY_Z1()

```



```

int NEWPAGE = 0;
if (NBUS > 3) NEWPAGE = 1;
clear_line( 1, 23);
for (D=0; D<2; ++D) {
    if ( D == 1 ) {
        if ( NEWPAGE ) clear_line( 1, 23);
        else printf( "\n\n" );
    }
    printf( "-----\n" );
    printf( "      Z1 MATRIX FOR ZONE %c\n", D + 65 );
    printf( "-----\n\n" );
    printf( "BUS   BUS           R           X\n\n" );
    for (k= i=0; i<NB[D]; ++i) {
        P = BusZone[D][i];
        for (j=i; j<NB[D]; ++j) {
            Q = BusZone[D][j];
            printf( FORM_5, P, Q, Z1[D][i][j] );
            if ( NEWPAGE ) see_more_data( &k, 7);
        }
    }
    if ( D == 0 )
        if ( NEWPAGE ) pause( "Press any key to see Z1 for zone B" );
}
pause( "Press any key to see Z2 for zone A" );
}

```

DISPLAY_Z2()

```

{
int bus, NEWPAGE = 0;
if (NBUS > 3) NEWPAGE = 1;
clear_line( 1, 23);
for (D=0; D<2; ++D) {
    if ( D == 1 ) {
        if ( NEWPAGE ) clear_line( 1, 23);
        else printf( "\n\n" );
    }
    printf( "-----\n" );
    printf( "      Z2 MATRIX FOR ZONE %c\n", D + 65 );
    printf( "-----\n\n" );
    printf( "BUS   TIE LINE           R           X\n\n" );
    for (k= i=0; i<NB[D]; ++i) {
        bus = BusZone[D][i];
        for (j=0; j<NLC; ++j) {
            P = bp[ CutLine[j] - 1 ];
            Q = bq[ CutLine[j] - 1 ];
            printf( FORM_6, bus, P, Q, Z2[D][i][j] );
            if ( NEWPAGE ) see_more_data( &k, 7);
        }
    }
    if ( D == 0 )

```

```

    if ( NEWPAGE ) pause( "Press any key to see Z2 for zone B" );
}
pause( "Press any key to see Y4" );
}

```

```

DISPLAY_Y4()
{

```

```

    clear_line( 1, 23);
    printf( "-----\n" );
    printf( "          Y4 MATRIX\n" );
    printf( "-----\n\n" );
    printf( "TIE LINE    TIE LINE          R          X\n\n" );
    for ( k = i=0; i<NLC; ++i) {
        P = bp[ CutLine[i] - 1 ];
        Q = bq[ CutLine[i] - 1 ];
        for ( j=i; j<NLC; ++j) {
            X = bp[ CutLine[j] - 1 ];
            Y = bq[ CutLine[j] - 1 ];
            printf( FORM_7, P, Q, X, Y, Y4[i][j] );
            see_more_data( &k, 7);
        }
    }
    printf( "\n" );
    printf( "-----\n" );
    printf( "Time used =%6.1f Sec.    Memory used =%5d Units\n", Time, Memo );
    printf( "-----\n" );
    pause( "Press any key to go back" );
}

```

```

/* ----- */

```

```

PRINT_Z1_Z2_Y4()
{

```

```

    if (( fx = fopen( "PRN", "w" )) != NULL ) {
        fprintf( fx, "%c%c", 27, 77 );
        PRINT_Z1();
        PRINT_Z2();
        PRINT_Y4();
        fclose( fx );
    }
    else error( "Incorrect Printer Name" );
}

```

```

PRINT_Z1()
{
    wait( "Printing . . .      Z1 MATRIX" );
    for (D=0; D<2; ++D) {
        fprintf( fx, "-----\n" );
        fprintf( fx, "      Z1 MATRIX FOR ZONE  %c\n", D + 65 );
        fprintf( fx, "-----\n\n" );
        fprintf( fx, "BUS   BUS           R           X\n\n" );
        for (i=0; i<NB[D]; ++i) {
            P = BusZone[D][i];
            for (j=i; j<NB[D]; ++j) {
                Q = BusZone[D][j];
                fprintf( fx, FORM_5, P, Q, Z1[D][i][j] );
            }
        }
        fprintf( fx, "\n-----\n\n\n" );
    }
}

```

```

PRINT_Z2()
{
    int bus;
    wait( "Printing . . .      Z2 MATRIX" );
    for (D=0; D<2; ++D) {
        fprintf( fx, "-----\n" );
        fprintf( fx, "      Z2 MATRIX FOR ZONE  %c\n", D + 65 );
        fprintf( fx, "-----\n\n" );
        fprintf( fx, "BUS   TIE LINE           R           X\n\n" );

        for (i=0; i<NB[D]; ++i) {
            bus = BusZone[D][i];
            for (j=0; j<NLC; ++j) {
                P = bp[ CutLine[j] - 1 ];
                Q = bq[ CutLine[j] - 1 ];
                fprintf( fx, FORM_6, bus, P, Q, Z2[D][i][j] );
            }
        }
        fprintf( fx, "\n-----\n\n\n" );
    }
}

```

```

PRINT_Y4()
{
    wait( "Printing . . .      Y4 MATRIX" );
    fprintf( fx, "-----\n" );
    fprintf( fx, "      Y4 MATRIX\n" );
}

```

```

fprintf( fx, "-----\n\n" );
fprintf( fx, "TIE LINE      TIE LINE      R      X \n\n" );
for (i=0; i<NLC; ++i) {
    P = bp[ CutLine[i] - 1 ];
    Q = bq[ CutLine[i] - 1 ];
    for (j=i; j<NLC; ++j) {
        X = bp[ CutLine[j] - 1 ];
        Y = bq[ CutLine[j] - 1 ];
        fprintf( fx, FORM_7, P, Q, X, Y, Y4[i][j] );
    }
}
fprintf( fx, "\n" );
fprintf( fx, "-----\n" );
fprintf( fx, "Time used =%5.1f Sec.      Memory used =%5d Units\n", Time, Memo );
fprintf( fx, "-----\n" );
fprintf( fx, "\f" );
}

```

```

/* ----- */

```

```

SAVE_Z1()
{
    strcpy( fileZ, filename );
    strcat( fileZ, "E" );
    if (( fx = fopen( fileZ, "w" )) != NULL ) {
        wait( "Saving Z1 file . . ." );
        for (D=0; D<2; ++D)
            for (i=0; i<NB[D]; ++i)
                for (j=i; j<NB[D]; ++j)
                    fprintf( fx, "%#20.17f %#20.17f\n", Z1[D][i][j] );
        fclose( fx );
    }
    else error( "Disk Drive not ready." ), ERR = 1;
}

```

```

LOAD_Z1()
{
    strcpy( fileZ, filename );
    strcat( fileZ, "E" );
    if (( fx = fopen( fileZ, "r" )) != NULL ) {
        wait( "Loading Z1 file . . ." );
        for (D=0; D<2; ++D)
            for (i=0; i<NB[D]; ++i)
                for (j=i; j<NB[D]; ++j)

```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย


```

        fscanf( fx, "%1f%1f\n", &Vx, &Vy ),
        Z1[D][i][j].x = Vx,
        Z1[D][i][j].y = Vy;
    fclose( fx );
    for (D=0; D<2; ++D)
        for (i=0; i<NB[D]; ++i)
            for (j=0; j<NB[D]; ++j)
                Z1[D][j][i] = Z1[D][i][j];
}
else ERR = 1;
}

SAVE_Z2()
{
    strcpy( fileZ, filename );
    strcat( fileZ, "F" );
    if (( fx = fopen( fileZ, "w" )) != NULL ) {
        wait( "Saving Z2 file . . ." );
        for (D=0; D<2; ++D)
            for (i=0; i<NB[D]; ++i)
                for (j=0; j<NLC; ++j)
                    fprintf( fx, "%#20.17f %#20.17f\n", Z2[D][i][j] );
        fclose( fx );
    }
    else error( "Disk Drive not ready." ), ERR = 1;
}

LOAD_Z2()
{
    strcpy( fileZ, filename );
    strcat( fileZ, "F" );
    if (( fx = fopen( fileZ, "r" )) != NULL ) {
        wait( "Loading Z2 file . . ." );
        for (D=0; D<2; ++D)
            for (i=0; i<NB[D]; ++i)
                for (j=0; j<NLC; ++j)
                    fscanf( fx, "%1f%1f\n", &Vx, &Vy ),
                    Z2[D][i][j].x = Vx,
                    Z2[D][i][j].y = Vy;
        fclose( fx );
    }
    else ERR = 1;
}

```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```

SAVE_Y4()
{
    strcpy( fileZ, filename );
    strcat( fileZ, "G" );
    if (( fx = fopen( fileZ, "w" )) != NULL ) {
        wait( "Saving Y4 file . . ." );
        for (i=0; i<NLC; ++i)
            for (j=i; j<NLC; ++j)
                fprintf( fx, "%#20.17f %#20.17f\n", Y4[i][j] );
        fprintf( fx, "%f\n", Time );
        fprintf( fx, "%d\n", Memo );
        fclose( fx );
    }
    else error( "Disk Drive not ready." ), ERR = 1;
}

```

```

LOAD_Y4()
{
    strcpy( fileZ, filename );
    strcat( fileZ, "G" );
    if (( fx = fopen( fileZ, "r" )) != NULL ) {
        wait( "Loading Y4 file . . ." );
        for (i=0; i<NLC; ++i)
            for (j=i; j<NLC; ++j)
                fscanf( fx, "%lf%lf\n", &Vx, &Vy ),
                    Y4[i][j].x = Vx,
                    Y4[i][j].y = Vy;
        fscanf( fx, "%f\n", &Time );
        fscanf( fx, "%d\n", &Memo );
        fclose( fx );
        for (i=0; i<NLC; ++i)
            for (j=i; j<NLC; ++j)
                Y4[j][i] = Y4[i][j];
    }
    else ERR = 1;
}

```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

/* ----- */

```

/* ----- */
/*
/*      File : P3_ZBUSC.c
/*
/*      ZBUS COLUMN Calculation routine for : FP1.EXE
/*
/* ----- */

#include <alloc.h>
#include <stdio.h>
#include <string.h>
#include <math.h>
#define COMPLEX struct complex

#include "MAX.B"
#include "FORM.B"

#define eif     else if

extern char    filename[15],
              BusZone[2][MBZ+1];
extern int     NB[2], NLC,
              ERR,
              NBUS,
              NLINE,
              bp[MREC],
              bq[MREC];

extern COMPLEX far    *Z1[2][MBZ+1];
extern COMPLEX        Z2[2][MBZ][MI],
                    Y4[MI][MI];

static int     bus, I, J,
              R, P, D, X;          /* Row, Column */

static char    BusList[MBUS+1];    /* Sum of bus */

static double  Vx, Vy,
              min = 0.00001;

static COMPLEX C00 = {0,0},
              Zsum, /* Z summation */
              Zt[MI], /* Z temporary */
              Zbc[MBUS], /* ZbusCol.temp.*/
              ZBC[MBUS]; /* ZBus Column */

static FILE    *fx;

COMPLEX far    *Z[MBUS+1];        /* Z-BUS matrix */

/* ----- */

```

```

SET_FAR_ZBUS()
{
    int i, j;
    long size2 = 16*(MBUS+1);
    for (i=0; i<=MBUS; ++i) {
        Z[i] = farmalloc( size2 );
        if ( Z[i] == NULL )
            error("Not enough room for Data"), exit(1);
    }
    for (i=0; i<=MBUS; ++i)
        for (j=0; j<=MBUS; ++j)
            Z[i][j].x = Z[i][j].y = 0;
}

/* ----- */

CALC_ZBUS_FORM_Z1_Z2_Y4()
{
    wait("Calculating . . . Z-BUS Loop count:"), set_count();
    strset( BusList, 0);
    strcpy( BusList, BusZone[A] );
    strcat( BusList, BusZone[B] );
    for (bus=1; bus<=NBUS; ++bus) {
        P = pos( bus, BusList );
        CALC_Y4_Z2T();
        CALC_Z2_Y4_Z2T();
        CALC_Z1_Z2_Y4_Z2T();
        TRANSFER_ZBC_TO_ZBUS();
    }
}

```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```

CALC_Y4_Z2T()
{
    int j;
    if (P < NB[A]) D = A, X = P;
    else D = B, X = P - NB[A];
    for (R=0; R<NLC; ++R) {
        plus_count();
        for (j=0; j<NLC; ++j)
            cmul( &Zt[j], Y4[R][j], Z2[D][X][j] );
        Zsum = C00;
        for (j=0; j<NLC; ++j)
            cadd( &Zsum, Zsum, Zt[j] );
    }
}

```

```

    Zbc[R] = Zsum;
}
}

```

```

CALC_Z2_Y4_Z2T()

```

```

{
    int j;
    for (R=0; R<NBUS; ++R) {
        plus_count();
        if (R < NB[A]) D = A, X = R;
        else          D = B, X = R - NB[A];
        for (j=0; j<NLC; ++j)
            cmul( &Zt[j], Z2[D][X][j], Zbc[j] );
        Zsum = C00;
        for (j=0; j<NLC; ++j)
            cadd( &Zsum, Zsum, Zt[j] );
        ZBC[R] = Zsum;
    }
}

```

```

CALC_Z1_Z2_Y4_Z2T()

```

```

{
    for (R=0; R<NBUS; ++R) {
        plus_count();
        if (R < NB[A]) {
            if (P < NB[A]) cstt( &ZBC[R], Z1[A][R][P], ZBC[R] );
            else          cstt( &ZBC[R], C00, ZBC[R] );
        }
        else {
            if (P < NB[A]) cstt( &ZBC[R], C00, ZBC[R] );
            else          cstt( &ZBC[R], Z1[B][R-NB[A]][P-NB[A]], ZBC[R] );
        }
    }
}

```

```

TRANSFER_ZBC_TO_ZBUS()

```

```

{
    plus_count();
    for (J=1; J<=NBUS; ++J) {
        R = pos( J, BusList );
        Z[bus][J] = ZBC[R];
    }
}

```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```
/* ----- */
```

```
DISPLAY_ZBUS()
```

```
{
    int k = 0;
    clear_line( 1, 23);
    printf( ZBUS_1 );
    printf( ZBUS_12 );
    printf( ZBUS_3 );
    printf( ZBUS_4 );
    for (I=1; I<=NBUS; ++I) {
        for (J=I; J<=NBUS; ++J) {
            Vx = fabs( Z[I][J].x );
            Vy = fabs( Z[I][J].y );
            if ( Vx < min && Vy < min ) printf( FORM_2, I, J );
            eif ( Vx < min )           printf( FORM_3, I, J, Z[I][J].y );
            eif ( Vy < min )           printf( FORM_4, I, J, Z[I][J].x );
            else                       printf( FORM_5, I, J, Z[I][J] );
            see_more_data( &k, 7);
        }
    }
    pause("press any key to go on");
}
```

```
PRINT_ZBUS()
```

```
{
    if (( fx = fopen("PRN","w")) != NULL) {
        wait("Printing . . . ZBUS");
        fprintf( fx, "%c%c", 27, 77);
        fprintf( fx, ZBUS_1 );
        fprintf( fx, ZBUS_12 );
        fprintf( fx, ZBUS_3 );
        fprintf( fx, ZBUS_4 );
        for (I=1; I<=NBUS; ++I) {
            for (J=I; J<=NBUS; ++J) {
                Vx = fabs( Z[I][J].x );
                Vy = fabs( Z[I][J].y );
                if ( Vx < min && Vy < min ) fprintf( fx, FORM_2, I, J );
                eif ( Vx < min )           fprintf( fx, FORM_3, I, J, Z[I][J].y );
                eif ( Vy < min )           fprintf( fx, FORM_4, I, J, Z[I][J].x );
                else                       fprintf( fx, FORM_5, I, J, Z[I][J] );
            }
        }
        fprintf( fx, "\f" );
        fclose( fx );
    }
}
```

```

else error("Incorrect Printer Name.");
}

```

```

SAVE_ZBUS()

```

```

{
char fileZ[15];
strcpy( fileZ, filename );
strcat( fileZ, "P" );
if (( fx = fopen( fileZ, "w" )) != NULL) {
wait("Saving Z-BUS file . . .");
for (I=1; I<=NBUS; ++I)
for (J=I; J<=NBUS; ++J)
fprintf( fx, "%#20.17f %#20.17f\n", Z[I][J] );
fclose( fx );
}
else error( "Disk Drive not ready." ), ERR = 1;
}

```

```

LOAD_ZBUS()

```

```

{
char fileZ[15];
strcpy( fileZ, filename );
strcat( fileZ, "P" );
if (( fx = fopen( fileZ, "r" )) != NULL) {
wait("Loading Z-BUS file . . .");
for (I=1; I<=NBUS; ++I) {
for (J=I; J<=NBUS; ++J) {
fscanf( fx, "%lf%lf\n", &Vx, &Vy);
Z[I][J].x = Vx;
Z[I][J].y = Vy;
}
}
fclose( fx );
for (I=1; I<=NBUS; ++I)
for (J=I; J<=NBUS; ++J)
Z[J][I] = Z[I][J];
}
else ERR = 1;
}

```

```

/* ----- */

```

```

/* ----- */
/*
/*   File : P4_FAULT.c
/*
/*   FAULT Calculation routine for : FP1.EXE
/*
/* ----- */

```

```

#include <stdio.h>
#include <math.h>
#define COMPLEX struct complex

#include "MAX.B"
#include "FORM.B"

#define eif else if

extern int Type, Phase, Bus, Quan,
           NBUS,
           NLINE,
           bp[MREC],
           bq[MREC];

extern COMPLEX z[MREC];
extern COMPLEX far *Z[MBUS+1];

static int P;
static COMPLEX TOP, BOT,
              C00 = {0,0},
              C10 = {1,0},
              C20 = {2,0},
              Ep = {1,0}, /* pre-fault voltage; bus fault */
              Ei = {1,0}, /* pre-fault voltage; other bus */
              zF = {0,0}, /* fault impedance */
              A1 = {-0.5, 0.866025}, /* temporary Complex variable */
              A2 = {-0.5,-0.866025}; /* temporary Complex variable */

static FILE *fx;

char buslink[MBL+1], /* bus link */
     BusLink[MBL+1]; /* Bus Links */

int NBL, NLL,
    bp1[MLL],
    bq1[MLL];
COMPLEX IB[3], /* I Bus fault (Sequence) */
        EB[3][MBL], /* V Bus fault (Sequence) */
        IL[3][MLL]; /* I Line fault (Sequence) */

/* ----- */

```

```
GOTO_FAULT(char State)
```

```
{
    int i;
```



```

if (Bus == 0) {
  for (Bus=1, i=0; i<NBUS; ++i, ++Bus) {
    FAULT( State );
  }
  Bus = 0;
}
elseif (Bus <= NBUS) {
  FAULT( State );
}
else error("Out off MAX. BUS");
}

```

```

FAULT(char State)
{
  wait("Calculating . . . FAULT SOLUTION   Loop count:"), set_count();
  P = Bus;
  SET_BUSLINK();
  BUS_CURRENT();
  BUS_VOLTAGE();
  LINE_CURRENT();
  if (Quan == 1) TRANSFER_TO_PHASE_QUANTITY();
  POLAR_OF_SOLUTION();
  if (State == 'D') DISPLAY_FAULT_SOLUTION();
  eif (State == 'P') PRINT_FAULT_SOLUTION();
}

```

```

/*-----*/

```

```

SET_BUSLINK()
{
  int i;
  strset( buslink, 0);
  for (i=0; i<NLINE; ++i)
    if (bp[i] == Bus || bq[i] == Bus)
      collect_buslink( bp[i], bq[i]);
  Sort_BusLink();
}

```

```

collect_buslink(char BP, char BQ)
{
  if (BP == 0) {
    if (!has( BQ, buslink)) strcat( buslink, &BQ, 1);
  }
}

```

ศูนย์วิทยทรัพยากร
มหาวิทยาลัย

```

else {
    if (!has( BP, buslink)) strncat( buslink, &BP, 1);
    if (!has( BQ, buslink)) strncat( buslink, &BQ, 1);
}
}

```

```

Sort_BusLink()
{
    int i;
    char I;
    strset( BusLink, 0);
    for (I=1, i=0; i<NBUS; ++i, ++I)
        if (has( I, buslink)) strncat( BusLink, &I, 1);
}

```

```

/* ----- */

```

```

BUS_CURRENT()
{
    if (Type == 0) THREE_PHASE();
    else LINE_TO_LINE();
}

```

```

THREE_PHASE()
{
    plus_count();
    IB[0] = IB[2] = C00; /* Ip(F)(0) & Ip(F)(2) */
    cadd( &BOT, zF, Z[P][P] );
    cdiv( &IB[1], Ep, BOT ); /* Ip(F)(1) */
}

```

```

LINE_TO_LINE()
{
    plus_count();
    IB[0] = C00; /* Ip(F)(0) = 0 */
    cmul( &TOP, C20, Z[P][P] );
    cadd( &BOT, TOP, zF );
    cdiv( &IB[1], Ep, BOT ); /* Ip(F)(1) */
    cstt( &IB[2], C00, IB[1] ); /* Ip(F)(2) */
}

```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



```

if (Phase == 1) {                               /* ref.phase B */
    cmul( &IB[2], A1, IB[1] );
    cstt( &IB[2], C00, IB[2] );
}
if (Phase == 2) {                               /* ref.phase C */
    cmul( &IB[2], A2, IB[1] );
    cstt( &IB[2], C00, IB[2] );
}
}

```

```

/* ----- */

```

```

BUS_VOLTAGE()
{
    int i, I, k = 1;
    for (I=1, i=0; i<NBUS; ++i, ++I) {
        if (has( I, BusLink)) {
            plus_count();
            cmul( &BOT, IB[0], C00 ), cstt( &EB[0][k], C00, BOT );
            cmul( &BOT, IB[1], Z[P][I] ), cstt( &EB[1][k], Ei, BOT );
            cmul( &BOT, IB[2], Z[P][I] ), cstt( &EB[2][k], C00, BOT );
            ++k;
        }
    }
    NBL = k;
}

```

```

/* ----- */

```

ศูนย์วิทยทรัพยากร

จุฬาลงกรณ์มหาวิทยาลัย

```

LINE_CURRENT()
{
    int x, k = 0;
    int bP, bQ, pP, pQ, swapPQ;
    for (x=0; x<NLINE; ++x) {
        if (bp[x] == P ;; bq[x] == P) {
            plus_count();
            bP = bp[x], bQ = bq[x];
            if (bP == P)
                swapPQ = bP, bP = bQ, bQ = swapPQ;

            pP = pos( bP, BusLink) + 1;
            pQ = pos( bQ, BusLink) + 1;
        }
    }
}

```

```

IL[0][k] = C00; /* IL(0)=0 */

if (bP == 0) cstt( &TOP, C10, EB[1][0] ); /* IL(1) */
else        cstt( &TOP, EB[1][pP], EB[1][pQ] );
cdiv( &IL[1][k], TOP, z[x] );

if (bP == 0) cstt( &TOP, EB[2][0], EB[2][pQ] ); /* IL(2) */
else        cstt( &TOP, EB[2][pP], EB[2][pQ] );
cdiv( &IL[2][k], TOP, z[x] );

bp1[k] = bP, bq1[k] = bQ;
++k;
}
}
NLL = k;
}
/* ----- */

```

```
TRANSFER_TO_PHASE_QUANTITY()
```

```

{
  int x, I;
  COMPLX T0, T1, T2, TA, TB, SM;
  plus_count();
  T0 = IB[0], T1 = IB[1], T2 = IB[2];
  cadd( &SM, T0, T1), cadd( &SM, SM, T2), IB[A] = SM;
  cmul( &TA, A2, T1), cmul( &TB, A1, T2);
  cadd( &SM, T0, TA), cadd( &SM, SM, TB), IB[B] = SM;
  cmul( &TA, A1, T1), cmul( &TB, A2, T2);
  cadd( &SM, T0, TA), cadd( &SM, SM, TB), IB[C] = SM;
  for (I=1; I<NBL; ++I) {
    plus_count();
    T0 = EB[0][I], T1 = EB[1][I], T2 = EB[2][I];
    cadd( &SM, T0, T1), cadd( &SM, SM, T2), EB[A][I] = SM;
    cmul( &TA, A2, T1), cmul( &TB, A1, T2);
    cadd( &SM, T0, TA), cadd( &SM, SM, TB), EB[B][I] = SM;
    cmul( &TA, A1, T1), cmul( &TB, A2, T2);
    cadd( &SM, T0, TA), cadd( &SM, SM, TB), EB[C][I] = SM;
  }
  for (x=0; x<NLL; ++x) {
    plus_count();
    T0 = IL[0][x], T1 = IL[1][x], T2 = IL[2][x];
    cadd( &SM, T0, T1), cadd( &SM, SM, T2), IL[A][x] = SM;
    cmul( &TA, A2, T1), cmul( &TB, A1, T2);
    cadd( &SM, T0, TA), cadd( &SM, SM, TB), IL[B][x] = SM;
    cmul( &TA, A1, T1), cmul( &TB, A2, T2);
    cadd( &SM, T0, TA), cadd( &SM, SM, TB), IL[C][x] = SM;
  }
}
}

```

```
/* ----- */
```

```
POLAR_OF_SOLUTION()
{
  int x, I;
  plus_count();
  polard( &IB[0] );
  polard( &IB[1] );
  polard( &IB[2] );
  for (I=1; I<NBL; ++I) {
    plus_count();
    polard( &EB[0][I] );
    polard( &EB[1][I] );
    polard( &EB[2][I] );
  }
  for (x=0; x<NLL; ++x) {
    plus_count();
    polard( &IL[0][x] );
    polard( &IL[1][x] );
    polard( &IL[2][x] );
  }
}
```

```
/* ----- */
```

```
DISPLAY_FAULT_SOLUTION()
{
  DISPLAY_OUTING();
  DISPLAY_BUS_CURRECT();
  DISPLAY_BUS_VOLTAGE();
  DISPLAY_LINE_CURRENT();
}
```

```
DISPLAY_OUTING()
{
  clear_line( 1, 23); setcur( 1, 1);
  printf( OUT___ );
  if (Type == 0) printf( OUT_00 , P, Phase+65 );
  else          printf( OUT_01 , P, Phase+65 );
  printf( OUT___ );
}
```

ศูนย์วิทยทรัพยากร
มหาวิทยาลัย

```

if (Quan == 0) printf("\nSEQUENCE QUANTITIES \n");
else          printf("\nPHASE QUANTITIES   \n");
}

```

```

DISPLAY_BUS_CURRENT()

```

```

{
printf("\nBUS CURRENT \n");
if (Quan == 0) printf( OUT_1 );
else          printf( OUT_1P );
printf( FORM_8 , P, IB[0], IB[1], IB[2]);
}

```

```

DISPLAY_BUS_VOLTAGE()

```

```

{
int I, k = 0;
printf("\nBUS VOLTAGE \n");
if (Quan == 0) printf( OUT_2 );
else          printf( OUT_2P );
for (I=1; I<NBL; ++I) {
printf( FORM_8 , BusLink[I-1], EB[0][I], EB[1][I], EB[2][I]);
see_more_data( &k, 13);
}
if ( NBL + NLL > 8 ) {
pause("press anykey to go on");
clear_line( 6, 23); setcur( 6, 1);
}
}

```

```

DISPLAY_LINE_CURRENT()

```

```

{
int x, k = 0;
printf("\nLINE CURRENT\n");
if (Quan == 0) printf( OUT_3 );
else          printf( OUT_3P );
for (x=0; x<NLL; ++x) {
printf( FORM_9 , bp1[x], bq1[x], IL[0][x], IL[1][x], IL[2][x]);
see_more_data( &k, 9);
}
pause("press anykey to go on");
}

```

```

/* ----- */

```

```
PRINT_FAULT_SOLUTION()
```

```
{
  if (( fx = fopen( "PRN", "w" )) != NULL ) {
    fprintf( fx, "%c%c", 27, 77 );
    PRINT_OUTING();
    PRINT_BUS_CURRECT();
    PRINT_BUS_VOLTAGE();
    PRINT_LINE_CURRENT();
    fclose( fx );
  }
  else error( "Incorrect Printer Name" );
}
```

```
PRINT_OUTING()
```

```
{
  wait( "Printing . . . FAULT SOLUTION" );
  fprintf( fx, OUT___ );
  if ( Type == 0 ) fprintf( fx, OUT_00 , P, Phase+65 );
  else             fprintf( fx, OUT_01 , P, Phase+65 );
  fprintf( fx, OUT___ );
  if ( Quan == 0 ) fprintf( fx, "\nSEQUENCE QUANTITIES\n" );
  else             fprintf( fx, "\nPHASE QUANTITIES\n" );
}
```

```
PRINT_BUS_CURRECT()
```

```
{
  wait( "Printing . . . BUS CURRENT" );
  fprintf( fx, "\nBUS CURRENT\n" );
  if ( Quan == 0 ) fprintf( fx, OUT_1 );
  else             fprintf( fx, OUT_1P );
  fprintf( fx, FORM_8 , P, IB[0], IB[1], IB[2] );
}
```

```
PRINT_BUS_VOLTAGE()
```

```
{
  int I;
  wait( "Printing . . . BUS VOLTAGE" );
  fprintf( fx, "\nBUS VOLTAGE\n" );
  if ( Quan == 0 ) fprintf( fx, OUT_2 );
  else             fprintf( fx, OUT_2P );
  for (I=1; I<NBL; ++I)
    fprintf( fx, FORM_8 , BusLink[I-1], EB[0][I], EB[1][I], EB[2][I] );
}
```

ศูนย์วิทยุโทรพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```
PRINT_LINE_CURRENT()  
{  
  int x;  
  wait( "Printing . . . LINE CURRENT" );  
  fprintf( fx, "\nLINE CURRENT\n" );  
  if (Quan == 0) fprintf( fx, OUT_3 );  
  else          fprintf( fx, OUT_3P );  
  for (x=0; x<NLL; ++x)  
    fprintf( fx, FORM_9 , bp1[x], bq1[x], IL[0][x], IL[1][x], IL[2][x] );  
  fprintf( fx, OUT___ );  
  fprintf( fx, "\n\n" );  
}  
  
/* ----- */
```



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

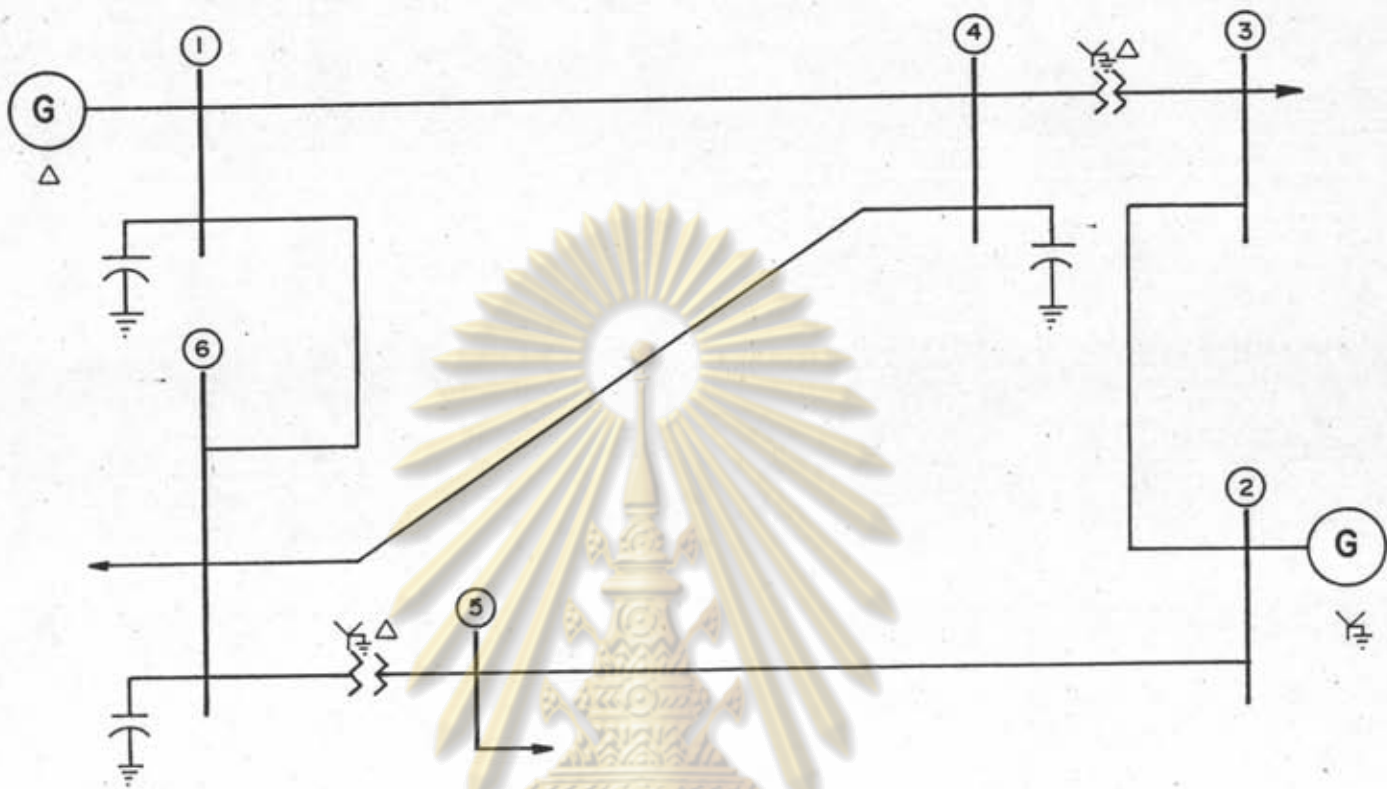


ภาคผนวก ง.

ตัวอย่างผลิตภัณฑ์ของระบบไฟฟ้ากำลัง ขนาด 6 บัส 9 สายส่ง

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ระบบไฟฟ้ากำลังตามมาตรฐาน IEEE ขนาด 6 บัส 9 สายส่ง



ข้อมูลระบบไฟฟ้ากำลัง

File: S2

Line	Bus		Equip.	Type	Self Impedance				Mutual Impedance		
	P	Q			Positive	Zero	Zm	Line			
1	0	1	GEN	Y	0.0200	0.2400	0.0000	0.0320			
2	0	2	GEN	YG	0.0300	0.4800	0.0000	0.0320			
3	1	4	LINE		0.1600	0.7400	0.8000	1.8500			
4	1	6	LINE		0.2460	1.0360	0.9840	2.0840			
5	2	3	LINE		1.4460	2.1000	3.7800	5.2600			
6	2	5	LINE		0.5640	1.2800	2.8200	3.8400			
7	3	4	XMER	D-YG	0.0000	0.2660	0.0000	0.2660			
8	6	4	LINE		0.1940	0.8140	0.9000	2.0600	0.5000	0.9500	3
9	5	6	XMER	D-YG	0.0000	0.6000	0.0000	0.6000			

BALANCE THREE PHASE FAULT AT BUS 1 REFERENCE PHASE : A

SEQUENCE QUANTITIES

BUS CURRENT		IO	ANGO	I1	ANG1	I2	ANG2
Bus							
1		0.0000	0.00	4.6251	-84.02	0.0000	0.00
BUS VOLTAGE		VO	ANGO	V1	ANG1	V2	ANG2
Bus							
1		0.0000	0.00	0.0000	0.00	0.0000	0.00
4		0.0000	0.00	0.1960	5.99	0.0000	0.00
6		0.0000	0.00	0.2378	1.27	0.0000	0.00
LINE CURRENT		IO	ANGO	I1	ANG1	I2	ANG2
Bus to bus							
0	1	0.0000	0.00	4.1523	-85.24	0.0000	0.00
4	1	0.0000	0.00	0.2588	-71.80	0.0000	0.00
6	1	0.0000	0.00	0.2233	-75.38	0.0000	0.00

BALANCE THREE PHASE FAULT AT BUS 1 REFERENCE PHASE : A

PHASE QUANTITIES

BUS CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus							
1		4.6251	-84.02	4.6251	155.98	4.6251	35.98
BUS VOLTAGE		VA	ANGA	VB	ANGB	VC	ANGC
Bus							
1		0.0000	0.00	0.0000	0.00	0.0000	0.00
4		0.1960	5.99	0.1960	-114.01	0.1960	125.99
6		0.2378	1.27	0.2378	-118.73	0.2378	121.27
LINE CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus to bus							
0	1	4.1523	-85.24	4.1523	154.76	4.1523	34.76
4	1	0.2588	-71.80	0.2588	168.20	0.2588	48.20
6	1	0.2233	-75.38	0.2233	164.62	0.2233	44.62



BALANCE THREE PHASE FAULT AT BUS 2 REFERENCE PHASE : A

SEQUENCE QUANTITIES

BUS CURRENT		I0	ANG0	I1	ANG1	I2	ANG2
Bus							
2		0.0000	0.00	2.6076	-83.38	0.0000	0.00

BUS VOLTAGE		V0	ANG0	V1	ANG1	V2	ANG2
Bus							
2		0.0000	0.00	0.0000	0.00	0.0000	0.00
3		0.0000	0.00	0.6043	-8.42	0.0000	0.00
5		0.0000	0.00	0.4328	-11.35	0.0000	0.00

LINE CURRENT		I0	ANG0	I1	ANG1	I2	ANG2
Bus to bus							
0	2	0.0000	0.00	2.0793	-86.42	0.0000	0.00
3	2	0.0000	0.00	0.2370	-63.87	0.0000	0.00
5	2	0.0000	0.00	0.3094	-77.57	0.0000	0.00

BALANCE THREE PHASE FAULT AT BUS 2 REFERENCE PHASE : A

PHASE QUANTITIES

BUS CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus							
2		2.6076	-83.38	2.6076	156.62	2.6076	36.62

BUS VOLTAGE		VA	ANGA	VB	ANGB	VC	ANGC
Bus							
2		0.0000	0.00	0.0000	0.00	0.0000	0.00
3		0.6043	-8.42	0.6043	-128.42	0.6043	111.58
5		0.4328	-11.35	0.4328	-131.35	0.4328	108.65

LINE CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus to bus							
0	2	2.0793	-86.42	2.0793	153.58	2.0793	33.58
3	2	0.2370	-63.87	0.2370	176.13	0.2370	56.13
5	2	0.3094	-77.57	0.3094	162.43	0.3094	42.43

 BALANCE THREE PHASE FAULT AT BUS 3 REFERENCE PHASE : A

SEQUENCE QUANTITIES

BUS CURRENT		I0	ANGO	I1	ANG1	I2	ANG2
Bus							
3		0.0000	0.00	1.3214	-77.61	0.0000	0.00

BUS VOLTAGE		V0	ANGO	V1	ANG1	V2	ANG2
Bus							
2		0.0000	0.00	0.8044	-4.64	0.0000	0.00
3		0.0000	0.00	0.0000	0.00	0.0000	0.00
4		0.0000	0.00	0.2726	7.08	0.0000	0.00

LINE CURRENT		I0	ANGO	I1	ANG1	I2	ANG2
Bus to bus							
2	3	0.0000	0.00	0.3155	-60.09	0.0000	0.00
4	3	0.0000	0.00	1.0250	-82.92	0.0000	0.00

 BALANCE THREE PHASE FAULT AT BUS 3 REFERENCE PHASE : A

PHASE QUANTITIES

BUS CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus							
3		1.3214	-77.61	1.3214	162.39	1.3214	42.39

BUS VOLTAGE		VA	ANGA	VB	ANGB	VC	ANGC
Bus							
2		0.8044	-4.64	0.8044	-124.64	0.8044	115.36
3		0.0000	0.00	0.0000	0.00	0.0000	0.00
4		0.2726	7.08	0.2726	-112.92	0.2726	127.08

LINE CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus to bus							
2	3	0.3155	-60.09	0.3155	179.91	0.3155	59.91
4	3	1.0250	-82.92	1.0250	157.08	1.0250	37.08

BALANCE THREE PHASE FAULT AT BUS 4 REFERENCE PHASE : A

SEQUENCE QUANTITIES

BUS CURRENT

Bus	I0	ANG0	I1	ANG1	I2	ANG2
4	0.0000	0.00	1.6892	-77.05	0.0000	0.00

BUS VOLTAGE

Bus	V0	ANG0	V1	ANG1	V2	ANG2
1	0.0000	0.00	0.7078	-2.29	0.0000	0.00
3	0.0000	0.00	0.0752	26.77	0.0000	0.00
4	0.0000	0.00	0.0000	0.00	0.0000	0.00
6	0.0000	0.00	0.4027	-2.59	0.0000	0.00

LINE CURRENT

Bus to bus	I0	ANG0	I1	ANG1	I2	ANG2
1 4	0.0000	0.00	0.9349	-80.09	0.0000	0.00
3 4	0.0000	0.00	0.2829	-63.23	0.0000	0.00
6 4	0.0000	0.00	0.4812	-79.19	0.0000	0.00

BALANCE THREE PHASE FAULT AT BUS 4 REFERENCE PHASE : A

PHASE QUANTITIES

BUS CURRENT

Bus	IA	ANGA	IB	ANGB	IC	ANGC
4	1.6892	-77.05	1.6892	162.95	1.6892	42.95

BUS VOLTAGE

Bus	VA	ANGA	VB	ANGB	VC	ANGC
1	0.7078	-2.29	0.7078	-122.29	0.7078	117.71
3	0.0752	26.77	0.0752	-93.23	0.0752	146.77
4	0.0000	0.00	0.0000	0.00	0.0000	0.00
6	0.4027	-2.59	0.4027	-122.59	0.4027	117.41

LINE CURRENT

Bus to bus	IA	ANGA	IB	ANGB	IC	ANGC
1 4	0.9349	-80.09	0.9349	159.91	0.9349	39.91
3 4	0.2829	-63.23	0.2829	176.77	0.2829	56.77
6 4	0.4812	-79.19	0.4812	160.81	0.4812	40.81

BALANCE THREE PHASE FAULT AT BUS 5 REFERENCE PHASE : A

SEQUENCE QUANTITIES

BUS CURRENT

Bus	I0	ANGO	I1	ANG1	I2	ANG2
5	0.0000	0.00	1.2146	-78.39	0.0000	0.00

BUS VOLTAGE

Bus	V0	ANGO	V1	ANG1	V2	ANG2
2	0.0000	0.00	0.7390	-4.88	0.0000	0.00
5	0.0000	0.00	0.0000	0.00	0.0000	0.00
6	0.0000	0.00	0.4163	6.06	0.0000	0.00

LINE CURRENT

Bus to bus	I0	ANGO	I1	ANG1	I2	ANG2
2 5	0.0000	0.00	0.5283	-71.10	0.0000	0.00
6 5	0.0000	0.00	0.6938	-83.94	0.0000	0.00

BALANCE THREE PHASE FAULT AT BUS 5 REFERENCE PHASE : A

PHASE QUANTITIES

BUS CURRENT

Bus	IA	ANGA	IB	ANGB	IC	ANGC
5	1.2146	-78.39	1.2146	161.61	1.2146	41.61

BUS VOLTAGE

Bus	VA	ANGA	VB	ANGB	VC	ANGC
2	0.7390	-4.88	0.7390	-124.88	0.7390	115.12
5	0.0000	0.00	0.0000	0.00	0.0000	0.00
6	0.4163	6.06	0.4163	-113.94	0.4163	126.06

LINE CURRENT

Bus to bus	IA	ANGA	IB	ANGB	IC	ANGC
2 5	0.5283	-71.10	0.5283	168.90	0.5283	48.90
6 5	0.6938	-83.94	0.6938	156.06	0.6938	36.06

BALANCE THREE PHASE FAULT AT BUS 6 REFERENCE PHASE : A

SEQUENCE QUANTITIES

BUS CURRENT		I0	ANGO	I1	ANG1	I2	ANG2
Bus							
6		0.0000	0.00	1.6002	-77.96	0.0000	0.00
BUS VOLTAGE		V0	ANGO	V1	ANG1	V2	ANG2
Bus							
1		0.0000	0.00	0.7380	-2.02	0.0000	0.00
4		0.0000	0.00	0.4337	-1.09	0.0000	0.00
5		0.0000	0.00	0.2334	12.91	0.0000	0.00
6		0.0000	0.00	0.0000	0.00	0.0000	0.00
LINE CURRENT		I0	ANGO	I1	ANG1	I2	ANG2
Bus to bus							
1	6	0.0000	0.00	0.6931	-78.66	0.0000	0.00
4	6	0.0000	0.00	0.5182	-77.68	0.0000	0.00
5	6	0.0000	0.00	0.3889	-77.09	0.0000	0.00

BALANCE THREE PHASE FAULT AT BUS 6 REFERENCE PHASE : A

PHASE QUANTITIES

BUS CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus							
6		1.6002	-77.96	1.6002	162.04	1.6002	42.04
BUS VOLTAGE		VA	ANGA	VB	ANGB	VC	ANGC
Bus							
1		0.7380	-2.02	0.7380	-122.02	0.7380	117.98
4		0.4337	-1.09	0.4337	-121.09	0.4337	118.91
5		0.2334	12.91	0.2334	-107.09	0.2334	132.91
6		0.0000	0.00	0.0000	0.00	0.0000	0.00
LINE CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus to bus							
1	6	0.6931	-78.66	0.6931	161.34	0.6931	41.34
4	6	0.5182	-77.68	0.5182	162.32	0.5182	42.32
5	6	0.3889	-77.09	0.3889	162.91	0.3889	42.91

LINE TO LINE FAULT AT BUS 1 REFERENCE PHASE : A

SEQUENCE QUANTITIES

BUS CURRENT

Bus	I0	ANGO	I1	ANG1	I2	ANG2
1	0.0000	0.00	2.3126	-84.02	2.3126	95.98

BUS VOLTAGE

Bus	V0	ANGO	V1	ANG1	V2	ANG2
1	0.0000	0.00	0.5000	0.00	0.5000	-0.00
4	0.0000	0.00	0.5975	0.98	0.4027	-1.46
6	0.0000	0.00	0.6189	0.24	0.3811	-0.40

LINE CURRENT

Bus to bus	I0	ANGO	I1	ANG1	I2	ANG2
0 1	0.0000	0.00	2.0761	-85.24	2.0761	94.76
4 1	0.0000	0.00	0.1294	-71.80	0.1294	108.20
6 1	0.0000	0.00	0.1117	-75.38	0.1117	104.62

LINE TO LINE FAULT AT BUS 1 REFERENCE PHASE : A

PHASE QUANTITIES

BUS CURRENT

Bus	IA	ANGA	IB	ANGB	IC	ANGC
1	0.0000	0.00	4.0055	-174.02	4.0055	5.98

BUS VOLTAGE

Bus	VA	ANGA	VB	ANGB	VC	ANGC
1	1.0000	-0.00	0.5000	180.00	0.5000	180.00
4	1.0000	-0.00	0.5110	-160.71	0.5445	161.94
6	1.0000	-0.00	0.5365	-157.43	0.5449	157.80

LINE CURRENT

Bus to bus	IA	ANGA	IB	ANGB	IC	ANGC
0 1	0.0000	0.00	3.5960	-175.24	3.5960	4.76
4 1	0.0000	0.00	0.2242	-161.80	0.2242	18.20
6 1	0.0000	0.00	0.1934	-165.38	0.1934	14.62

 LINE TO LINE FAULT AT BUS 2 REFERENCE PHASE : A

SEQUENCE QUANTITIES

BUS CURRENT		I0	ANGO	I1	ANG1	I2	ANG2
Bus							
2		0.0000	0.00	1.3038	-83.38	1.3038	96.62

BUS VOLTAGE		V0	ANGO	V1	ANG1	V2	ANG2
Bus							
2		0.0000	0.00	0.5000	0.00	0.5000	-0.00
3		0.0000	0.00	0.8001	-3.17	0.2059	12.41
5		0.0000	0.00	0.7134	-3.42	0.2910	8.41

LINE CURRENT		I0	ANGO	I1	ANG1	I2	ANG2
Bus to bus							
0	2	0.0000	0.00	1.0396	-86.42	1.0396	93.58
3	2	0.0000	0.00	0.1185	-63.87	0.1185	116.13
5	2	0.0000	0.00	0.1547	-77.57	0.1547	102.43

 LINE TO LINE FAULT AT BUS 2 REFERENCE PHASE : A

PHASE QUANTITIES

BUS CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus							
2		0.0000	0.00	2.2582	-173.38	2.2582	6.62

BUS VOLTAGE		VA	ANGA	VB	ANGB	VC	ANGC
Bus							
2		1.0000	-0.00	0.5000	180.00	0.5000	180.00
3		1.0000	-0.00	0.7750	-138.08	0.6688	129.27
5		1.0000	-0.00	0.6813	-147.36	0.5628	139.23

LINE CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus to bus							
0	2	0.0000	0.00	1.8007	-176.42	1.8007	3.58
3	2	0.0000	0.00	0.2053	-153.87	0.2053	26.13
5	2	0.0000	0.00	0.2680	-167.57	0.2680	12.43

 LINE TO LINE FAULT AT BUS 3 REFERENCE PHASE : A

SEQUENCE QUANTITIES

BUS CURRENT

Bus	I0	ANGO	I1	ANG1	I2	ANG2
3	0.0000	0.00	0.6607	-77.61	0.6607	102.39

BUS VOLTAGE

Bus	V0	ANGO	V1	ANG1	V2	ANG2
2	0.0000	0.00	0.9015	-2.07	0.1043	18.19
3	0.0000	0.00	0.5000	0.00	0.5000	-0.00
4	0.0000	0.00	0.6355	1.52	0.3651	-2.64

LINE CURRENT

Bus to bus	I0	ANGO	I1	ANG1	I2	ANG2
2 3	0.0000	0.00	0.1577	-60.09	0.1577	119.91
4 3	0.0000	0.00	0.5125	-82.92	0.5125	97.08

 LINE TO LINE FAULT AT BUS 3 REFERENCE PHASE : A

PHASE QUANTITIES

BUS CURRENT

Bus	IA	ANGA	IB	ANGB	IC	ANGC
3	0.0000	0.00	1.1444	-167.61	1.1444	12.39

BUS VOLTAGE

Bus	VA	ANGA	VB	ANGB	VC	ANGC
2	1.0000	-0.00	0.8898	-128.71	0.8239	122.57
3	1.0000	-0.00	0.5000	180.00	0.5000	180.00
4	1.0000	-0.00	0.5260	-153.55	0.5787	156.11

LINE CURRENT

Bus to bus	IA	ANGA	IB	ANGB	IC	ANGC
2 3	0.0000	0.00	0.2732	-150.09	0.2732	29.91
4 3	0.0000	0.00	0.8877	-172.92	0.8877	7.08

 LINE TO LINE FAULT AT BUS 4 REFERENCE PHASE : A

SEQUENCE QUANTITIES

BUS CURRENT

Bus	I0	ANGO	I1	ANG1	I2	ANG2
4	0.0000	0.00	0.8446	-77.05	0.8446	102.95

BUS VOLTAGE

Bus	V0	ANGO	V1	ANG1	V2	ANG2
1	0.0000	0.00	0.8537	-0.95	0.1471	5.51
3	0.0000	0.00	0.5339	1.82	0.4667	-2.08
4	0.0000	0.00	0.5000	0.00	0.5000	-0.00
6	0.0000	0.00	0.7012	-0.74	0.2990	1.74

LINE CURRENT

Bus to bus	I0	ANGO	I1	ANG1	I2	ANG2
1 4	0.0000	0.00	0.4674	-80.09	0.4674	99.91
3 4	0.0000	0.00	0.1414	-63.23	0.1414	116.77
6 4	0.0000	0.00	0.2406	-79.19	0.2406	100.81

 LINE TO LINE FAULT AT BUS 4 REFERENCE PHASE : A

PHASE QUANTITIES

BUS CURRENT

Bus	IA	ANGA	IB	ANGB	IC	ANGC
4	0.0000	0.00	1.4629	-167.05	1.4629	12.95

BUS VOLTAGE

Bus	VA	ANGA	VB	ANGB	VC	ANGC
1	1.0000	0.00	0.8064	-130.57	0.7754	127.83
3	1.0000	-0.00	0.4742	-172.95	0.5325	173.73
4	1.0000	-0.00	0.5000	180.00	0.5000	180.00
6	1.0000	-0.00	0.6224	-145.96	0.5965	144.27

LINE CURRENT

Bus to bus	IA	ANGA	IB	ANGB	IC	ANGC
1 4	0.0000	0.00	0.8096	-170.09	0.8096	9.91
3 4	0.0000	0.00	0.2450	-153.23	0.2450	26.77
6 4	0.0000	0.00	0.4168	-169.19	0.4168	10.81

 LINE TO LINE FAULT AT BUS 5 REFERENCE PHASE : A

SEQUENCE QUANTITIES

BUS CURRENT		I0	ANGO	I1	ANG1	I2	ANG2
Bus							
5		0.0000	0.00	0.6073	-78.39	0.6073	101.61
BUS VOLTAGE		V0	ANGO	V1	ANG1	V2	ANG2
Bus							
2		0.0000	0.00	0.8687	-2.07	0.1355	13.40
5		0.0000	0.00	0.5000	0.00	0.5000	-0.00
6		0.0000	0.00	0.7073	1.78	0.2939	-4.29
LINE CURRENT		I0	ANGO	I1	ANG1	I2	ANG2
Bus to bus							
2	5	0.0000	0.00	0.2642	-71.10	0.2642	108.90
6	5	0.0000	0.00	0.3469	-83.94	0.3469	96.06

 LINE TO LINE FAULT AT BUS 5 REFERENCE PHASE : A

PHASE QUANTITIES

BUS CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus							
5		0.0000	0.00	1.0518	-168.39	1.0518	11.61
BUS VOLTAGE		VA	ANGA	VB	ANGB	VC	ANGC
Bus							
2		1.0000	-0.00	0.8450	-131.00	0.7779	124.94
5		1.0000	-0.00	0.5000	180.00	0.5000	180.00
6		1.0000	-0.00	0.5847	-142.19	0.6465	146.33
LINE CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus to bus							
2	5	0.0000	0.00	0.4576	-161.10	0.4576	18.90
6	5	0.0000	0.00	0.6008	-173.94	0.6008	6.06



 LINE TO LINE FAULT AT BUS 6 REFERENCE PHASE : A

SEQUENCE QUANTITIES

BUS CURRENT

Bus	I0	ANGO	I1	ANG1	I2	ANG2
6	0.0000	0.00	0.8001	-77.96	0.8001	102.04

BUS VOLTAGE

Bus	V0	ANGO	V1	ANG1	V2	ANG2
1	0.0000	0.00	0.8689	-0.86	0.1319	5.66
4	0.0000	0.00	0.7168	-0.33	0.2832	0.83
5	0.0000	0.00	0.6143	2.43	0.3872	-3.86
6	0.0000	0.00	0.5000	0.00	0.5000	-0.00

LINE CURRENT

Bus to bus	I0	ANGO	I1	ANG1	I2	ANG2
1 6	0.0000	0.00	0.3466	-78.66	0.3466	101.34
4 6	0.0000	0.00	0.2591	-77.68	0.2591	102.32
5 6	0.0000	0.00	0.1945	-77.09	0.1945	102.91

 LINE TO LINE FAULT AT BUS 6 REFERENCE PHASE : A

PHASE QUANTITIES

BUS CURRENT

Bus	IA	ANGA	IB	ANGB	IC	ANGC
6	0.0000	0.00	1.3858	-167.96	1.3858	12.04

BUS VOLTAGE

Bus	VA	ANGA	VB	ANGB	VC	ANGC
1	1.0000	0.00	0.8252	-129.28	0.7975	126.78
4	1.0000	-0.00	0.6310	-143.48	0.6196	142.70
5	1.0000	-0.00	0.4957	-156.58	0.5796	160.13
6	1.0000	-0.00	0.5000	180.00	0.5000	180.00

LINE CURRENT

Bus to bus	IA	ANGA	IB	ANGB	IC	ANGC
1 6	0.0000	0.00	0.6002	-168.66	0.6002	11.34
4 6	0.0000	0.00	0.4488	-167.68	0.4488	12.32
5 6	0.0000	0.00	0.3368	-167.09	0.3368	12.91

ศูนย์วิจัยทรัพยากรน้ำ
 จุฬาลงกรณ์มหาวิทยาลัย

SINGLE LINE TO GROUND FAULT AT BUS 1 REFERENCE PHASE : A

SEQUENCE QUANTITIES

BUS CURRENT		I0	ANGO	I1	ANG1	I2	ANG2
Bus							
1		0.6263	-75.16	0.6263	-75.16	0.6263	-75.16
BUS VOLTAGE		V0	ANGO	V1	ANG1	V2	ANG2
Bus							
1		0.7336	176.74	0.8665	-1.38	0.1354	-171.14
4		0.0722	-159.58	0.8920	-0.90	0.1091	-172.60
6		0.2143	-169.39	0.8980	-0.97	0.1032	-171.54
LINE CURRENT		I0	ANGO	I1	ANG1	I2	ANG2
Bus to bus							
0	1	0.0000	0.00	0.5623	-76.38	0.5623	-76.38
4	1	0.3973	-76.08	0.0350	-62.95	0.0350	-62.95
6	1	0.2291	-73.56	0.0302	-66.52	0.0302	-66.52

SINGLE LINE TO GROUND FAULT AT BUS 1 REFERENCE PHASE : A

PHASE QUANTITIES

BUS CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus							
1		1.8789	-75.16	0.0000	0.00	0.0000	0.00
BUS VOLTAGE		VA	ANGA	VB	ANGB	VC	ANGC
Bus							
1		0.0000	0.00	1.3611	-143.82	1.4385	139.79
4		0.7180	-4.26	0.9902	-117.65	0.9706	118.26
6		0.5893	-6.80	1.0784	-124.35	1.0387	125.86
LINE CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus to bus							
0	1	1.1245	-76.38	0.5623	103.62	0.5623	103.62
4	1	0.4658	-74.12	0.3633	-77.33	0.3633	-77.33
6	1	0.2893	-72.09	0.1992	-74.63	0.1992	-74.63

SINGLE LINE TO GROUND FAULT AT BUS 2 REFERENCE PHASE : A

SEQUENCE QUANTITIES

BUS CURRENT		I0	ANGO	I1	ANG1	I2	ANG2
Bus							
2		1.2519	-83.64	1.2519	-83.64	1.2519	-83.64
BUS VOLTAGE		V0	ANGO	V1	ANG1	V2	ANG2
Bus							
2		0.0401	-173.64	0.5199	0.24	0.4801	179.74
3		0.0401	-173.64	0.8078	-2.95	0.1977	-167.85
5		0.0401	-173.64	0.7245	-3.13	0.2794	-171.85
LINE CURRENT		I0	ANGO	I1	ANG1	I2	ANG2
Bus to bus							
0	2	1.2519	-83.64	0.9983	-86.69	0.9983	-86.69
3	2	0.0000	0.00	0.1138	-64.14	0.1138	-64.14
5	2	0.0000	0.00	0.1486	-77.83	0.1486	-77.83

SINGLE LINE TO GROUND FAULT AT BUS 2 REFERENCE PHASE : A

PHASE QUANTITIES

BUS CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus							
2		3.7557	-83.64	0.0000	0.00	0.0000	0.00
BUS VOLTAGE		VA	ANGA	VB	ANGB	VC	ANGC
Bus							
2		0.0000	0.00	0.8747	-93.91	0.8614	93.98
3		0.5803	-8.69	0.8984	-112.69	0.9674	110.99
5		0.4156	-11.61	0.8716	-107.58	0.9389	106.28
LINE CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus to bus							
0	2	3.2473	-85.51	0.2605	-71.90	0.2605	-71.90
3	2	0.2276	-64.14	0.1138	115.86	0.1138	115.86
5	2	0.2971	-77.83	0.1486	102.17	0.1486	102.17

SINGLE LINE TO GROUND FAULT AT BUS 3 REFERENCE PHASE : A

SEQUENCE QUANTITIES

BUS CURRENT		I0	ANGO	I1	ANG1	I2	ANG2
Bus	3	0.1263	-58.77	0.1263	-58.77	0.1263	-58.77
BUS VOLTAGE		V0	ANGO	V1	ANG1	V2	ANG2
Bus	2	0.0040	-148.77	0.9841	-0.70	0.0199	-142.98
	3	0.8214	175.69	0.9101	-1.94	0.0956	-161.17
	4	0.0000	0.00	0.9332	-1.20	0.0698	-163.80
LINE CURRENT		I0	ANGO	I1	ANG1	I2	ANG2
Bus to bus	2 3	0.1263	-58.77	0.0302	-41.26	0.0302	-41.26
	4 3	0.0000	0.00	0.0980	-64.08	0.0980	-64.08

SINGLE LINE TO GROUND FAULT AT BUS 3 REFERENCE PHASE : A

PHASE QUANTITIES

BUS CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus	3	0.3789	-58.77	0.0000	0.00	0.0000	0.00
BUS VOLTAGE		VA	ANGA	VB	ANGB	VC	ANGC
Bus	2	0.9650	-1.55	0.9852	-119.66	1.0025	119.10
	3	0.0000	0.00	1.4518	-147.81	1.5583	142.04
	4	0.8668	-2.57	0.9509	-117.09	0.9857	116.06
LINE CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus to bus	2 3	0.1847	-53.13	0.0980	-64.08	0.0980	-64.08
	4 3	0.1959	-64.08	0.0980	115.92	0.0980	115.92

SINGLE LINE TO GROUND FAULT AT BUS 4 REFERENCE PHASE : A

SEQUENCE QUANTITIES

BUS CURRENT

Bus	I0	ANGO	I1	ANG1	I2	ANG2
4	0.7034	-78.93	0.7034	-78.93	0.7034	-78.93

BUS VOLTAGE

Bus	V0	ANGO	V1	ANG1	V2	ANG2
1	0.0811	-163.36	0.8778	-0.51	0.1225	-176.37
3	0.0000	0.00	0.6128	2.51	0.3887	176.04
4	0.1699	-170.73	0.5840	1.34	0.4164	178.12
6	0.0409	-151.85	0.7510	0.05	0.2490	179.86

LINE CURRENT

Bus to bus	I0	ANGO	I1	ANG1	I2	ANG2
1 4	0.0182	-59.31	0.3893	-81.97	0.3893	-81.97
3 4	0.0000	0.00	0.1178	-65.12	0.1178	-65.12
6 4	0.0500	-62.78	0.2004	-81.07	0.2004	-81.07
0 4	0.6386	-80.73	0.0000	0.00	0.0000	0.00

SINGLE LINE TO GROUND FAULT AT BUS 4 REFERENCE PHASE : A

PHASE QUANTITIES

BUS CURRENT

Bus	IA	ANGA	IB	ANGB	IC	ANGC
4	2.1101	-78.93	0.0000	0.00	0.0000	0.00

BUS VOLTAGE

Bus	VA	ANGA	VB	ANGB	VC	ANGC
1	0.6790	-3.27	0.9922	-117.32	0.9648	118.17
3	0.2308	13.46	0.8999	-97.17	0.8466	97.62
4	0.0000	0.00	0.9413	-105.49	0.8624	106.95
6	0.4663	-2.22	0.9312	-107.95	0.8935	108.74

LINE CURRENT

Bus to bus	IA	ANGA	IB	ANGB	IC	ANGC
1 4	0.7954	-81.47	0.3726	96.95	0.3726	96.95
3 4	0.2356	-65.12	0.1178	114.88	0.1178	114.88
6 4	0.4485	-79.07	0.1537	93.07	0.1537	93.07
0 4	0.6386	-80.73	0.6386	-80.73	0.6386	-80.73

SINGLE LINE TO GROUND FAULT AT BUS 5 REFERENCE PHASE : A

SEQUENCE QUANTITIES

BUS CURRENT		I0	ANG0	I1	ANG1	I2	ANG2
Bus							
5		0.1581	-60.12	0.1581	-60.12	0.1581	-60.12
BUS VOLTAGE		V0	ANG0	V1	ANG1	V2	ANG2
Bus							
2		0.0051	-150.12	0.9702	-1.09	0.0353	-148.33
5		0.7572	173.81	0.8774	-2.67	0.1302	-161.73
6		-0.0000	0.00	0.9260	-1.14	0.0765	-166.02
LINE CURRENT		I0	ANG0	I1	ANG1	I2	ANG2
Bus to bus							
2	5	0.1581	-60.12	0.0688	-52.83	0.0688	-52.83
6	5	0.0000	0.00	0.0903	-65.67	0.0903	-65.67

SINGLE LINE TO GROUND FAULT AT BUS 5 REFERENCE PHASE : A

PHASE QUANTITIES

BUS CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus							
5		0.4742	-60.12	0.0000	0.00	0.0000	0.00
BUS VOLTAGE		VA	ANGA	VB	ANGB	VC	ANGC
Bus							
2		0.9364	-2.42	0.9734	-119.16	1.0015	118.27
5		0.0000	0.00	1.3521	-146.63	1.5007	138.80
6		0.8523	-2.49	0.9485	-116.67	0.9816	115.70
LINE CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus to bus							
2	5	0.2950	-56.73	0.0903	-65.67	0.0903	-65.67
6	5	0.1806	-65.67	0.0903	114.33	0.0903	114.33

SINGLE LINE TO GROUND FAULT AT BUS 6 REFERENCE PHASE : A

SEQUENCE QUANTITIES

BUS CURRENT		I0	ANGO	I1	ANG1	I2	ANG2
Bus							
6		0.5803	-80.00	0.5803	-80.00	0.5803	-80.00
BUS VOLTAGE		V0	ANGO	V1	ANG1	V2	ANG2
Bus							
1		0.1985	-174.24	0.9046	-0.38	0.0956	-176.38
4		0.0337	-152.93	0.7946	0.31	0.2054	178.79
5		0.0000	0.00	0.7213	2.29	0.2808	174.10
6		0.2764	-174.64	0.6377	1.16	0.3626	177.96
LINE CURRENT		I0	ANGO	I1	ANG1	I2	ANG2
Bus to bus							
1	6	0.0338	-60.38	0.2514	-80.70	0.2514	-80.70
4	6	0.0930	-63.85	0.1879	-79.72	0.1879	-79.72
5	6	0.0000	0.00	0.1410	-79.13	0.1410	-79.13
0	6	0.4606	-84.64	0.0000	0.00	0.0000	0.00

SINGLE LINE TO GROUND FAULT AT BUS 6 REFERENCE PHASE : A

PHASE QUANTITIES

BUS CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus							
6		1.7409	-80.00	0.0000	0.00	0.0000	0.00
BUS VOLTAGE		VA	ANGA	VB	ANGB	VC	ANGC
Bus							
1		0.6124	-3.00	1.0662	-124.38	1.0434	125.24
4		0.5593	-0.68	0.9433	-110.13	0.9065	110.99
5		0.4451	7.45	0.9217	-103.85	0.8658	104.77
6		0.0000	0.00	0.9945	-114.52	0.9245	116.52
LINE CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus to bus							
1	6	0.5345	-79.45	0.2200	96.24	0.2200	96.24
4	6	0.4660	-76.59	0.1017	85.79	0.1017	85.79
5	6	0.2821	-79.13	0.1410	100.87	0.1410	100.87
0	6	0.4606	-84.64	0.4606	-84.64	0.4606	-84.64

DOUBLE LINE TO GROUND FAULT AT BUS 1 REFERENCE PHASE : A

SEQUENCE QUANTITIES

BUS CURRENT		I0	ANG0	I1	ANG1	I2	ANG2
Bus							
1		0.3915	107.08	2.5049	-83.16	2.1208	94.96

BUS VOLTAGE		V0	ANG0	V1	ANG1	V2	ANG2
Bus							
1		0.4585	-1.02	0.4585	-1.02	0.4585	-1.02
4		0.0451	22.66	0.5639	0.46	0.3693	-2.47
6		0.1339	12.85	0.5872	-0.33	0.3495	-1.41

LINE CURRENT			I0	ANG0	I1	ANG1	I2	ANG2
Bus to bus								
0	1		0.0000	0.00	2.2488	-84.37	1.9040	93.75
4	1		0.2483	106.16	0.1402	-70.94	0.1187	107.18
6	1		0.1432	108.68	0.1210	-74.51	0.1024	103.61

DOUBLE LINE TO GROUND FAULT AT BUS 1 REFERENCE PHASE : A

PHASE QUANTITIES

BUS CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus							
1		0.0000	0.00	4.1586	178.02	3.9348	14.40

BUS VOLTAGE		VA	ANGA	VB	ANGB	VC	ANGC
Bus							
1		1.3756	-1.02	0.0000	0.00	0.0000	0.00
4		0.9745	0.35	0.4323	-160.31	0.4823	156.56
6		1.0673	0.96	0.3741	-152.95	0.4190	144.77

LINE CURRENT			IA	ANGA	IB	ANGB	IC	ANGC
Bus to bus								
0	1		0.3515	-74.14	3.6339	-177.96	3.5663	7.54
4	1		0.2271	104.91	0.3381	148.22	0.3469	66.48
6	1		0.1245	107.61	0.2556	158.09	0.2368	54.60

DOUBLE LINE TO GROUND FAULT AT BUS 2 REFERENCE PHASE : A

SEQUENCE QUANTITIES

BUS CURRENT		I0	ANGO	I1	ANG1	I2	ANG2
Bus							
2		2.2365	95.68	2.4220	-83.81	0.1866	102.30
BUS VOLTAGE		V0	ANGO	V1	ANG1	V2	ANG2
Bus							
2		0.0716	5.68	0.0716	5.68	0.0716	5.68
3		0.0716	5.68	0.6308	-7.23	0.0295	18.09
5		0.0716	5.68	0.4707	-9.17	0.0416	14.09
LINE CURRENT		I0	ANGO	I1	ANG1	I2	ANG2
Bus to bus							
0	2	2.2365	95.68	1.9313	-86.86	0.1488	99.25
3	2	0.0000	0.00	0.2201	-64.31	0.0170	121.80
5	2	0.0000	0.00	0.2874	-78.00	0.0221	108.11

DOUBLE LINE TO GROUND FAULT AT BUS 2 REFERENCE PHASE : A

PHASE QUANTITIES

BUS CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus							
2		0.0000	0.00	4.0129	129.92	4.0748	62.03
BUS VOLTAGE		VA	ANGA	VB	ANGB	VC	ANGC
Bus							
2		0.2147	5.68	0.0000	0.00	0.0000	0.00
3		0.7278	-4.98	0.5802	-124.95	0.5878	107.73
5		0.5792	-5.73	0.4155	-127.88	0.4210	104.81
LINE CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus to bus							
0	2	0.4654	107.42	3.6420	124.43	3.5749	64.57
3	2	0.2033	-64.82	0.2275	179.60	0.2305	52.28
5	2	0.2654	-78.51	0.2970	165.90	0.3010	38.59

DOUBLE LINE TO GROUND FAULT AT BUS 3 REFERENCE PHASE : A

SEQUENCE QUANTITIES

BUS CURRENT		I0	ANGO	I1	ANG1	I2	ANG2
Bus							
3		0.0730	124.29	0.6947	-76.48	0.6270	101.15
BUS VOLTAGE		V0	ANGO	V1	ANG1	V2	ANG2
Bus							
2		0.0023	34.29	0.8972	-2.32	0.0990	16.94
3		0.4745	-1.24	0.4745	-1.24	0.4745	-1.24
4		0.0000	0.00	0.6163	0.94	0.3465	-3.88
LINE CURRENT		I0	ANGO	I1	ANG1	I2	ANG2
Bus to bus							
2	3	0.0730	124.29	0.1659	-58.97	0.1497	118.66
4	3	0.0000	0.00	0.5388	-81.80	0.4863	95.84

DOUBLE LINE TO GROUND FAULT AT BUS 3 REFERENCE PHASE : A

PHASE QUANTITIES

BUS CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus							
3		0.0000	0.00	1.1895	-172.50	1.1082	17.65
BUS VOLTAGE		VA	ANGA	VB	ANGB	VC	ANGC
Bus							
2		0.9931	-0.35	0.8819	-128.59	0.8248	122.02
3		1.4235	-1.24	0.0000	0.00	0.0000	0.00
4		0.9620	-0.79	0.5060	-153.26	0.5641	154.71
LINE CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus to bus							
2	3	0.0566	118.98	0.2934	-166.09	0.2765	46.91
4	3	0.0566	-61.02	0.8986	-174.59	0.8775	8.80

DOUBLE LINE TO GROUND FAULT AT BUS 4 REFERENCE PHASE : A

SEQUENCE QUANTITIES

BUS CURRENT		I0	ANGO	I1	ANG1	I2	ANG2
Bus							
4		0.9346	97.94	1.3107	-78.83	0.3813	109.10

BUS VOLTAGE		V0	ANGO	V1	ANG1	V2	ANG2
Bus							
1		0.1077	13.52	0.7724	-1.10	0.0664	11.66
3		0.0000	0.00	0.2816	9.98	0.2107	4.07
4		0.2257	6.15	0.2257	6.15	0.2257	6.15
6		0.0543	25.02	0.5360	0.03	0.1350	7.89

LINE CURRENT		I0	ANGO	I1	ANG1	I2	ANG2
Bus to bus							
1	4	0.0241	117.56	0.7254	-81.87	0.2110	106.06
3	4	0.0000	0.00	0.2195	-65.02	0.0639	122.91
6	4	0.0664	114.10	0.3734	-80.97	0.1086	106.96
0	4	0.8485	96.15	0.0000	0.00	0.0000	0.00

DOUBLE LINE TO GROUND FAULT AT BUS 4 REFERENCE PHASE : A

PHASE QUANTITIES

BUS CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus							
4		0.0000	0.00	1.9357	146.78	2.1127	54.33

BUS VOLTAGE		VA	ANGA	VB	ANGB	VC	ANGC
Bus							
1		0.9423	1.44	0.6772	-119.98	0.7009	114.39
3		0.4916	7.45	0.2325	-157.21	0.2744	174.50
4		0.6771	6.15	0.0000	0.00	0.0000	0.00
6		0.7201	3.33	0.4505	-131.99	0.4515	126.71

LINE CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus to bus							
1	4	0.4951	-86.17	0.8425	170.45	0.8713	28.70
3	4	0.1565	-68.25	0.2506	-171.36	0.2636	43.97
6	4	0.2043	-90.05	0.4644	165.68	0.4578	36.34
0	4	0.8485	96.15	0.8485	96.15	0.8485	96.15

DOUBLE LINE TO GROUND FAULT AT BUS 5 REFERENCE PHASE : A

SEQUENCE QUANTITIES

BUS CURRENT		I0	ANGO	I1	ANG1	I2	ANG2
Bus							
5		0.0968	124.18	0.6522	-76.76	0.5629	99.72
BUS VOLTAGE		V0	ANGO	V1	ANG1	V2	ANG2
Bus							
2		0.0031	34.18	0.8603	-2.52	0.1256	11.51
5		0.4635	-1.89	0.4635	-1.89	0.4635	-1.89
6		0.0000	0.00	0.6849	1.22	0.2724	-6.18
LINE CURRENT		I0	ANGO	I1	ANG1	I2	ANG2
Bus to bus							
2	5	0.0968	124.18	0.2837	-69.47	0.2449	107.01
6	5	0.0000	0.00	0.3725	-82.31	0.3215	94.17

DOUBLE LINE TO GROUND FAULT AT BUS 5 REFERENCE PHASE : A

PHASE QUANTITIES

BUS CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus							
5		0.0000	0.00	1.1156	-175.29	1.0051	19.27
BUS VOLTAGE		VA	ANGA	VB	ANGB	VC	ANGC
Bus							
2		0.9851	-0.64	0.8315	-130.78	0.7782	123.92
5		1.3904	-1.89	0.0000	0.00	0.0000	0.00
6		0.9557	-0.88	0.5627	-141.39	0.6324	144.66
LINE CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus to bus							
2	5	0.0553	118.63	0.5039	-174.03	0.4387	33.80
6	5	0.0553	-61.37	0.6119	-176.33	0.5908	8.54

DOUBLE LINE TO GROUND FAULT AT BUS 6 REFERENCE PHASE : A

SEQUENCE QUANTITIES

BUS CURRENT		I0	ANGO	I1	ANG1	I2	ANG2
Bus	6	0.6352	97.57	1.1170	-79.23	0.4841	104.97
BUS VOLTAGE		V0	ANGO	V1	ANG1	V2	ANG2
Bus	1	0.2173	3.33	0.8166	-0.99	0.0798	8.59
	4	0.0369	24.64	0.6046	0.29	0.1714	3.76
	5	0.0000	0.00	0.4642	5.98	0.2342	-0.93
	6	0.3025	2.93	0.3025	2.93	0.3025	2.93
LINE CURRENT		I0	ANGO	I1	ANG1	I2	ANG2
Bus to bus							
1	6	0.0370	117.19	0.4838	-79.93	0.2097	104.27
4	6	0.1018	113.72	0.3618	-78.95	0.1568	105.25
5	6	0.0000	0.00	0.2715	-78.36	0.1177	105.84
0	6	0.5042	92.93	0.0000	0.00	0.0000	0.00

DOUBLE LINE TO GROUND FAULT AT BUS 6 REFERENCE PHASE : A

PHASE QUANTITIES

BUS CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus	6	0.0000	0.00	1.6194	156.12	1.7419	45.09
BUS VOLTAGE		VA	ANGA	VB	ANGB	VC	ANGC
Bus	1	1.1123	0.54	0.6744	-112.06	0.6849	107.70
	4	0.8097	2.10	0.5152	-134.54	0.5175	132.13
	5	0.6973	3.66	0.3738	-144.09	0.4302	156.04
	6	0.9076	2.93	0.0000	0.00	0.0000	0.00
LINE CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus to bus							
1	6	0.2408	-86.19	0.6240	175.27	0.6252	27.38
4	6	0.1113	-96.64	0.5034	168.62	0.4813	37.18
5	6	0.1544	-81.56	0.3396	179.81	0.3514	25.56
0	6	0.5042	92.93	0.5042	92.93	0.5042	92.93

SINGLE LINE TO GROUND FAULT AT BUS 1 REFERENCE PHASE : B

SEQUENCE QUANTITIES

BUS CURRENT		I0	ANGO	I1	ANG1	I2	ANG2
Bus							
1		0.6263	164.84	0.6263	-75.16	0.6263	44.84
BUS VOLTAGE		V0	ANGO	V1	ANG1	V2	ANG2
Bus							
1		0.7336	56.74	1.0901	-125.55	0.1354	-51.14
4		0.0722	80.42	1.0698	-124.65	0.1091	-52.60
6		0.2143	70.61	1.0673	-124.34	0.1032	-51.54
LINE CURRENT		I0	ANGO	I1	ANG1	I2	ANG2
Bus to bus							
0	1	0.0000	0.00	7.7190	-56.74	0.5623	43.62
4	1	0.3973	163.92	0.0350	-62.95	0.0350	57.05
6	1	0.2291	166.44	0.0302	-66.52	0.0302	53.48

SINGLE LINE TO GROUND FAULT AT BUS 1 REFERENCE PHASE : B

PHASE QUANTITIES

BUS CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus							
1		0.0000	0.00	1.8789	164.84	0.0000	0.00
BUS VOLTAGE		VA	ANGA	VB	ANGB	VC	ANGC
Bus							
1		0.4062	-111.14	1.7320	90.00	1.4385	19.79
4		1.0405	-120.61	1.2082	109.54	0.9706	-1.74
6		0.8918	-121.56	1.3087	105.65	1.0387	5.86
LINE CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus to bus							
0	1	7.6379	-52.59	8.2508	-178.05	7.2997	60.40
4	1	0.3633	162.67	0.4658	165.88	0.3633	162.67
6	1	0.1992	165.37	0.2893	167.91	0.1992	165.37

SINGLE LINE TO GROUND FAULT AT BUS 2 REFERENCE PHASE : B

SEQUENCE QUANTITIES

BUS CURRENT

Bus	I0	ANGO	I1	ANG1	I2	ANG2
2	1.2519	156.36	1.2519	-83.64	1.2519	36.36

BUS VOLTAGE

Bus	V0	ANGO	V1	ANG1	V2	ANG2
2	0.0401	66.36	1.3064	-138.61	0.4801	-60.26
3	0.0401	66.36	1.1421	-127.37	0.1977	-47.85
5	0.0401	66.36	1.1930	-130.61	0.2794	-51.85

LINE CURRENT

Bus to bus	I0	ANGO	I1	ANG1	I2	ANG2
0 2	1.2519	156.36	4.4919	-62.85	0.9983	33.31
3 2	0.0000	0.00	0.1138	-64.14	0.1138	55.86
5 2	0.0000	0.00	0.1486	-77.83	0.1486	42.17

SINGLE LINE TO GROUND FAULT AT BUS 2 REFERENCE PHASE : B

PHASE QUANTITIES

BUS CURRENT

Bus	IA	ANGA	IB	ANGB	IC	ANGC
2	0.0000	0.00	3.7557	156.36	0.0000	0.00

BUS VOLTAGE

Bus	VA	ANGA	VB	ANGB	VC	ANGC
2	1.4403	-120.26	1.7320	90.00	0.8614	-26.02
3	1.1541	-118.15	1.3295	105.83	0.9674	-9.01
5	1.2372	-118.37	1.4479	100.99	0.9389	-13.72

LINE CURRENT

Bus to bus	IA	ANGA	IB	ANGB	IC	ANGC
0 2	3.4206	-59.49	6.6298	169.80	3.5450	67.66
3 2	0.1138	-4.14	0.2276	175.86	0.1138	-4.14
5 2	0.1486	-17.83	0.2971	162.17	0.1486	-17.83

SINGLE LINE TO GROUND FAULT AT BUS 3 REFERENCE PHASE : B

SEQUENCE QUANTITIES

BUS CURRENT		I0	ANGO	I1	ANG1	I2	ANG2
Bus							
3		0.1263	-178.77	0.1263	-58.77	0.1263	61.23
BUS VOLTAGE		V0	ANGO	V1	ANG1	V2	ANG2
Bus							
2		0.0040	91.23	1.0184	-120.44	0.0199	-22.98
3		0.8214	55.69	1.0738	-123.36	0.0956	-41.17
4		0.0000	0.00	1.0515	-122.63	0.0698	-43.80
LINE CURRENT		I0	ANGO	I1	ANG1	I2	ANG2
Bus to bus							
2	3	0.1263	-178.77	0.0302	-41.26	0.0302	78.74
4	3	0.0000	0.00	0.0980	-64.08	0.0980	55.92

SINGLE LINE TO GROUND FAULT AT BUS 3 REFERENCE PHASE : B

PHASE QUANTITIES

BUS CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus							
3		0.0000	0.00	0.3789	-178.77	0.0000	0.00
BUS VOLTAGE		VA	ANGA	VB	ANGB	VC	ANGC
Bus							
2		1.0125	-119.44	1.0404	119.04	1.0025	-0.90
3		0.2867	-101.17	1.7320	90.00	1.5583	22.04
4		1.0672	-118.95	1.1050	114.98	0.9857	-3.94
LINE CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus to bus							
2	3	0.0980	175.92	0.1847	-173.13	0.0980	175.92
4	3	0.0980	-4.08	0.1959	175.92	0.0980	-4.08

SINGLE LINE TO GROUND FAULT AT BUS 4 REFERENCE PHASE : B

SEQUENCE QUANTITIES

BUS CURRENT		I0	ANG0	I1	ANG1	I2	ANG2
Bus							
4		0.7034	161.07	0.7034	-78.93	0.7034	41.07
BUS VOLTAGE		V0	ANG0	V1	ANG1	V2	ANG2
Bus							
1		0.0811	76.64	1.0727	-125.46	0.1225	-56.37
3		0.0000	0.00	1.2216	-136.61	0.3887	-63.96
4		0.1699	69.27	1.2513	-137.07	0.4164	-61.88
6		0.0409	88.15	1.1445	-130.88	0.2490	-60.14
LINE CURRENT		I0	ANG0	I1	ANG1	I2	ANG2
Bus to bus							
1	4	0.0182	-179.31	0.3893	-81.97	0.3893	38.03
3	4	0.0000	0.00	0.1178	-65.12	0.1178	54.88
6	4	0.0500	177.22	0.2004	-81.07	0.2004	38.93
0	4	0.6386	159.27	0.0000	0.00	0.0000	0.00

SINGLE LINE TO GROUND FAULT AT BUS 4 REFERENCE PHASE : B

PHASE QUANTITIES

BUS CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus							
4		0.0000	0.00	2.1101	161.07	0.0000	0.00
BUS VOLTAGE		VA	ANGA	VB	ANGB	VC	ANGC
Bus							
1		1.0447	-120.85	1.2225	107.74	0.9648	-1.83
3		1.3880	-121.11	1.5122	92.49	0.8466	-22.38
4		1.2492	-121.88	1.7320	90.00	0.8624	-13.05
6		1.2131	-120.94	1.3605	100.53	0.8935	-11.26
LINE CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus to bus							
1	4	0.3726	-23.05	0.7954	158.53	0.3726	-23.05
3	4	0.1178	-5.12	0.2356	174.88	0.1178	-5.12
6	4	0.1537	-26.93	0.4485	160.93	0.1537	-26.93
0	4	0.6386	159.27	0.6386	159.27	0.6386	159.27



SINGLE LINE TO GROUND FAULT AT BUS 5 REFERENCE PHASE : B

SEQUENCE QUANTITIES

BUS CURRENT		I0	ANGO	I1	ANG1	I2	ANG2
Bus							
5		0.1581	179.88	0.1581	-60.12	0.1581	59.88

BUS VOLTAGE		V0	ANGO	V1	ANG1	V2	ANG2
Bus							
2		0.0051	89.88	1.0312	-120.93	0.0353	-28.33
5		0.7572	53.81	1.1005	-124.51	0.1302	-41.73
6		0.0000	0.00	1.0546	-122.99	0.0765	-46.02

LINE CURRENT		I0	ANGO	I1	ANG1	I2	ANG2
Bus to bus							
2	5	0.1581	179.88	0.0688	-52.83	0.0688	67.17
6	5	0.0000	0.00	0.0903	-65.67	0.0903	54.33

SINGLE LINE TO GROUND FAULT AT BUS 5 REFERENCE PHASE : B

PHASE QUANTITIES

BUS CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus							
5		0.0000	0.00	0.4742	179.88	0.0000	0.00

BUS VOLTAGE		VA	ANGA	VB	ANGB	VC	ANGC
Bus							
2		1.0258	-119.11	1.0671	118.07	1.0015	-1.73
5		0.3905	-101.73	1.7320	90.00	1.5007	18.80
6		1.0744	-119.01	1.1117	114.32	0.9816	-4.30

LINE CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus to bus							
2	5	0.0903	174.33	0.2950	-176.73	0.0903	174.33
6	5	0.0903	-5.67	0.1806	174.33	0.0903	-5.67

SINGLE LINE TO GROUND FAULT AT BUS 6 REFERENCE PHASE : B

SEQUENCE QUANTITIES

BUS CURRENT

Bus	I0	ANGO	I1	ANG1	I2	ANG2
6	0.5803	160.00	0.5803	-80.00	0.5803	40.00

BUS VOLTAGE

Bus	V0	ANGO	V1	ANG1	V2	ANG2
1	0.1985	65.76	1.0560	-124.33	0.0956	-56.38
4	0.0337	87.07	1.1136	-129.30	0.2054	-61.21
5	0.0000	0.00	1.1437	-132.95	0.2808	-65.90
6	0.2764	65.36	1.2131	-135.31	0.3626	-62.04

LINE CURRENT

Bus to bus	I0	ANGO	I1	ANG1	I2	ANG2
1 6	0.0338	179.62	0.2514	-80.70	0.2514	39.30
4 6	0.0930	176.15	0.1879	-79.72	0.1879	40.28
5 6	0.0000	0.00	0.1410	-79.13	0.1410	40.87
0 6	0.4606	155.36	0.0000	0.00	0.0000	0.00

SINGLE LINE TO GROUND FAULT AT BUS 6 REFERENCE PHASE : B

PHASE QUANTITIES

BUS CURRENT

Bus	IA	ANGA	IB	ANGB	IC	ANGC
6	0.0000	0.00	1.7409	160.00	0.0000	0.00

BUS VOLTAGE

Bus	VA	ANGA	VB	ANGB	VC	ANGC
1	0.8980	-120.89	1.2632	105.31	1.0434	5.24
4	1.1755	-120.96	1.2832	102.85	0.9065	-9.01
5	1.2796	-121.29	1.3319	97.36	0.8658	-15.23
6	1.0879	-122.04	1.7320	90.00	0.9245	-3.48

LINE CURRENT

Bus to bus	IA	ANGA	IB	ANGB	IC	ANGC
1 6	0.2200	-23.76	0.5345	160.55	0.2200	-23.76
4 6	0.1017	-34.21	0.4660	163.41	0.1017	-34.21
5 6	0.1410	-19.13	0.2821	160.87	0.1410	-19.13
0 6	0.4606	155.36	0.4606	155.36	0.4606	155.36



SINGLE LINE TO GROUND FAULT AT BUS 1 REFERENCE PHASE : C

SEQUENCE QUANTITIES

BUS CURRENT		I0	ANGO	I1	ANG1	I2	ANG2
Bus							
1		0.6263	44.84	0.6263	-75.16	0.6263	164.84

BUS VOLTAGE		V0	ANGO	V1	ANG1	V2	ANG2
Bus							
1		0.7336	-63.26	1.0564	126.87	0.1354	68.86
4		0.0722	-39.58	1.0468	125.52	0.1091	67.40
6		0.2143	-49.39	1.0423	125.29	0.1032	68.46

LINE CURRENT		I0	ANGO	I1	ANG1	I2	ANG2
Bus to bus							
0	1	0.0000	0.00	7.6379	-112.59	0.5623	163.62
4	1	0.3973	43.92	0.0350	-62.95	0.0350	177.05
6	1	0.2291	46.44	0.0302	-66.52	0.0302	173.48

SINGLE LINE TO GROUND FAULT AT BUS 1 REFERENCE PHASE : C

PHASE QUANTITIES

BUS CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus							
1		0.0000	0.00	0.0000	0.00	1.8789	44.84

BUS VOLTAGE		VA	ANGA	VB	ANGB	VC	ANGC
Bus							
1		0.4062	128.86	1.3611	-23.82	1.7320	-90.00
4		1.0405	119.39	0.9902	2.35	1.1292	-106.03
6		0.8918	118.44	1.0784	-4.35	1.2128	-101.03

LINE CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus to bus							
0	1	7.7190	-116.74	7.1270	129.23	8.0984	9.76
4	1	0.3633	42.67	0.3633	42.67	0.4658	45.88
6	1	0.1992	45.37	0.1992	45.37	0.2893	47.91

SINGLE LINE TO GROUND FAULT AT BUS 2 REFERENCE PHASE : C

SEQUENCE QUANTITIES

BUS CURRENT		I0	ANGO	I1	ANG1	I2	ANG2
Bus							
2		1.2519	36.36	1.2519	-83.64	1.2519	156.36
BUS VOLTAGE		V0	ANGO	V1	ANG1	V2	ANG2
Bus							
2		0.0401	-53.64	1.3094	138.46	0.4801	59.74
3		0.0401	-53.64	1.0772	130.06	0.1977	72.15
5		0.0401	-53.64	1.1340	133.22	0.2794	68.15
LINE CURRENT		I0	ANGO	I1	ANG1	I2	ANG2
Bus to bus							
0	2	1.2519	36.36	4.4956	-110.10	0.9983	153.31
3	2	0.0000	0.00	0.1138	-64.14	0.1138	175.86
5	2	0.0000	0.00	0.1486	-77.83	0.1486	162.17

SINGLE LINE TO GROUND FAULT AT BUS 2 REFERENCE PHASE : C

PHASE QUANTITIES

BUS CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus							
2		0.0000	0.00	0.0000	0.00	3.7557	36.36
BUS VOLTAGE		VA	ANGA	VB	ANGB	VC	ANGC
Bus							
2		1.4403	119.74	0.8747	26.09	1.7320	-90.00
3		1.1541	121.85	0.8984	7.31	1.2100	-100.04
5		1.2372	121.63	0.8716	12.42	1.3441	-95.60
LINE CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus to bus							
0	2	3.3511	-115.24	3.6754	119.64	6.6018	18.21
3	2	0.1138	-124.14	0.1138	-124.14	0.2276	55.86
5	2	0.1486	-137.83	0.1486	-137.83	0.2971	42.17

SINGLE LINE TO GROUND FAULT AT BUS 3 REFERENCE PHASE : C

SEQUENCE QUANTITIES

BUS CURRENT

Bus	I0	ANGO	I1	ANG1	I2	ANG2
3	0.1263	61.23	0.1263	-58.77	0.1263	-178.77

BUS VOLTAGE

Bus	V0	ANGO	V1	ANG1	V2	ANG2
2	0.0040	-28.77	0.9978	121.14	0.0199	97.02
3	0.8214	-64.31	1.0228	125.26	0.0956	78.83
4	0.0000	0.00	1.0189	123.81	0.0698	76.20

LINE CURRENT

Bus to bus	I0	ANGO	I1	ANG1	I2	ANG2
2 3	0.1263	61.23	0.0302	-41.26	0.0302	-161.26
4 3	0.0000	0.00	0.0980	-64.08	0.0980	175.92

SINGLE LINE TO GROUND FAULT AT BUS 3 REFERENCE PHASE : C

PHASE QUANTITIES

BUS CURRENT

Bus	IA	ANGA	IB	ANGB	IC	ANGC
3	0.0000	0.00	0.0000	0.00	0.3789	61.23

BUS VOLTAGE

Bus	VA	ANGA	VB	ANGB	VC	ANGC
2	1.0125	120.56	0.9852	0.34	0.9960	-117.49
3	0.2867	138.83	1.4518	-27.81	1.7320	-90.00
4	1.0672	121.05	0.9509	2.91	1.0422	-112.53

LINE CURRENT

Bus to bus	IA	ANGA	IB	ANGB	IC	ANGC
2 3	0.0980	55.92	0.0980	55.92	0.1847	66.87
4 3	0.0980	-124.08	0.0980	-124.08	0.1959	55.92

SINGLE LINE TO GROUND FAULT AT BUS 4 REFERENCE PHASE : C

SEQUENCE QUANTITIES

BUS CURRENT		I0	ANGO	I1	ANG1	I2	ANG2
Bus							
4		0.7034	41.07	0.7034	-78.93	0.7034	161.07

BUS VOLTAGE		V0	ANGO	V1	ANG1	V2	ANG2
Bus							
1		0.0811	-43.36	1.0601	125.94	0.1225	63.63
3		0.0000	0.00	1.2591	134.83	0.3887	56.04
4		0.1699	-50.73	1.2701	136.16	0.4164	58.12
6		0.0409	-31.85	1.1454	130.84	0.2490	59.86

LINE CURRENT		I0	ANGO	I1	ANG1	I2	ANG2
Bus to bus							
1	4	0.0182	60.69	0.3893	-81.97	0.3893	158.03
3	4	0.0000	0.00	0.1178	-65.12	0.1178	174.88
6	4	0.0500	57.22	0.2004	-81.07	0.2004	158.93
0	4	0.6386	39.27	0.0000	0.00	0.0000	0.00

SINGLE LINE TO GROUND FAULT AT BUS 4 REFERENCE PHASE : C

PHASE QUANTITIES

BUS CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus							
4		0.0000	0.00	0.0000	0.00	2.1101	41.07

BUS VOLTAGE		VA	ANGA	VB	ANGB	VC	ANGC
Bus							
1		1.0447	119.15	0.9922	2.68	1.1663	-105.18
3		1.3880	118.89	0.8999	22.83	1.5725	-95.80
4		1.2492	118.12	0.9413	14.51	1.7320	-90.00
6		1.2131	119.06	0.9312	12.05	1.3373	-99.35

LINE CURRENT		IA	ANGA	IB	ANGB	IC	ANGC
Bus to bus							
1	4	0.3726	-143.05	0.3726	-143.05	0.7954	38.53
3	4	0.1178	-125.12	0.1178	-125.12	0.2356	54.88
6	4	0.1537	-146.93	0.1537	-146.93	0.4485	40.93
0	4	0.6386	39.27	0.6386	39.27	0.6386	39.27

SINGLE LINE TO GROUND FAULT AT BUS 5 REFERENCE PHASE : C

SEQUENCE QUANTITIES

BUS CURRENT

Bus	I0	ANGO	I1	ANG1	I2	ANG2
5	0.1581	59.88	0.1581	-60.12	0.1581	179.88

BUS VOLTAGE

Bus	V0	ANGO	V1	ANG1	V2	ANG2
2	0.0051	-30.12	0.9996	122.02	0.0353	91.67
5	0.7572	-66.19	1.0343	127.08	0.1302	78.27
6	0.0000	0.00	1.0238	124.12	0.0765	73.98

LINE CURRENT

Bus to bus	I0	ANGO	I1	ANG1	I2	ANG2
2 5	0.1581	59.88	0.0688	-52.83	0.0688	-172.83
6 5	0.0000	0.00	0.0903	-65.67	0.0903	174.33

SINGLE LINE TO GROUND FAULT AT BUS 5 REFERENCE PHASE : C

PHASE QUANTITIES

BUS CURRENT

Bus	IA	ANGA	IB	ANGB	IC	ANGC
5	0.0000	0.00	0.0000	0.00	0.4742	59.88

BUS VOLTAGE

Bus	VA	ANGA	VB	ANGB	VC	ANGC
2	1.0258	120.89	0.9734	0.84	1.0008	-115.67
5	0.3905	138.27	1.3521	-26.63	1.7320	-90.00
6	1.0744	120.99	0.9485	3.33	1.0525	-111.97

LINE CURRENT

Bus to bus	IA	ANGA	IB	ANGB	IC	ANGC
2 5	0.0903	54.33	0.0903	54.33	0.2950	63.27
6 5	0.0903	-125.67	0.0903	-125.67	0.1806	54.33

SINGLE LINE TO GROUND FAULT AT BUS 6 REFERENCE PHASE : C

SEQUENCE QUANTITIES

BUS CURRENT

Bus	I0	ANGO	I1	ANG1	I2	ANG2
6	0.5803	40.00	0.5803	-80.00	0.5803	160.00

BUS VOLTAGE

Bus	V0	ANGO	V1	ANG1	V2	ANG2
1	0.1985	-54.24	1.0460	124.70	0.0956	63.62
4	0.0337	-32.93	1.1203	129.02	0.2054	58.79
5	0.0000	0.00	1.1867	131.05	0.2808	54.10
6	0.2764	-54.64	1.2314	134.46	0.3626	57.96

LINE CURRENT

Bus to bus	I0	ANGO	I1	ANG1	I2	ANG2
1 6	0.0338	59.62	0.2514	-80.70	0.2514	159.30
4 6	0.0930	56.15	0.1879	-79.72	0.1879	160.28
5 6	0.0000	0.00	0.1410	-79.13	0.1410	160.87
0 6	0.4606	35.36	0.0000	0.00	0.0000	0.00

SINGLE LINE TO GROUND FAULT AT BUS 6 REFERENCE PHASE : C

PHASE QUANTITIES

BUS CURRENT

Bus	IA	ANGA	IB	ANGB	IC	ANGC
6	0.0000	0.00	0.0000	0.00	1.7409	40.00

BUS VOLTAGE

Bus	VA	ANGA	VB	ANGB	VC	ANGC
1	0.8980	119.11	1.0662	-4.38	1.2186	-103.19
4	1.1755	119.04	0.9433	9.87	1.2742	-102.41
5	1.2796	118.71	0.9217	16.15	1.4050	-101.11
6	1.0879	117.96	0.9945	5.48	1.7320	-90.00

LINE CURRENT

Bus to bus	IA	ANGA	IB	ANGB	IC	ANGC
1 6	0.2200	-143.76	0.2200	-143.76	0.5345	40.55
4 6	0.1017	-154.21	0.1017	-154.21	0.4660	43.41
5 6	0.1410	-139.13	0.1410	-139.13	0.2821	40.87
0 6	0.4606	35.36	0.4606	35.36	0.4606	35.36



ประวัติผู้เขียน

นายหนา ชาดะวราหะ เกิดเมื่อวันที่ 19 มกราคม พ.ศ. 2503 ณ จังหวัด นครราชสีมา สำเร็จการศึกษาระดับปริญญาตรี สาขาวิศวกรรมศาสตร์บัณฑิต จากคณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้า วิทยาเขตเจ้าคุณทหารลาดกระบัง เมื่อปี พ.ศ. 2526 หลังจากสำเร็จการศึกษาแล้วได้เข้าทำงานที่

- . หน่วยงานส่วนจำกัด ศิริกิจ อีเลคตริค ปี พ.ศ. 2526 ถึง พ.ศ. 2528 ในตำแหน่ง วิศวกรสนาม ติดตั้งระบบไฟฟ้าและสื่อสาร
- . บริษัท พี.เอ็ม.ไอ. จำกัด ปี พ.ศ. 2528 ถึง พ.ศ. 2530 ในตำแหน่งวิศวกรสนาม ควบคุมดูแลการติดตั้งระบบไฟฟ้าและสื่อสาร
- . บริษัท เอสคอน จำกัด ปี พ.ศ. 2531 จนถึงปัจจุบัน ในตำแหน่งวิศวกรไฟฟ้า ออกแบบระบบไฟฟ้าและสื่อสาร



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย