เอกสารอ้างอิง

1. Anderson, P.M. and Found, A.A., "Power System Control and Stability", Iowa State University Press, 1977
2. Byerly, R.T., Sherman, D.E. and Benon, R.J., "Eigenvalue Analysis of Synchronizing Power Flow Oscillations in Large Electric Power System", IEEE Transaction on Power Apparatus and System, Vol. PAS-101, No.1, pp. 235-248, January, 1982
3. Arjona, J.D., "Control of Self-Excited Oscillations in Power System", University of Florida, 1980
4. Yuan-Yih Hsu, Sheng-Wehn Shyue and Chung-Ching Su, "Low Frequency Oscillations in Longitudinal Power System : Experience with Dynamic of Taiwan Power System", IEEE Transaction on Power Systems, Vol. PWRS-2, pp. 92-100, No.1, February, 1987
5. Jansen, J.F., "Spontaneous Oscillations in Florida Power Grid", University of Florida, 1979
6. Byerly, R.T., Sherman, D.E. and Benon, R.J., "Frequency Domain Analysis of Low Frequency Oscillations", EPRI, El-726, Part 2, Project 744-1, April, 1978
7. IEEE Committee Report, "Dynamic Model for Steam and Hydro Turbines in Power System Studies", IEEE Transaction on Power Apparatus and System, November/December, 1973 pp. 92-100, No.1, February, 1987
8. Stagg, G.W. and El-abaid, "Computer Methods in Power System Analysis", New York, McGraw-Hill,1968
9. Shleif, F.R., Hunkins, H.D., Hattan, E.E. and Gish., W., "Control of Rotating Exciter for Power System Damping : Pilot Application and Experience", IEEE Transaction on Power Apparatus and System, Vol. PAS-88, pp. 1259-1266, August, 1969

10. Klopfenstien, A., "Experience with System Stabilizing Excitation Control on the Generation of the Southern California Adison Company", IEEE Transaction on Power Apparatus and System, Vol.PAS-90 pp. 698-706, March/April, 1971

11. DeMello,F.P. and Concordia, C., "Concept of Synchronous Machine Stability as Affected by Excitation Control", IEEE Transaction on Power Apparatus and System, Vol. PAS-88, pp. 316-329, April, 1969

12. DeMello,F.P. and Laskowski, T.F, "Concept Power System Dynamic Stability" IEEE Transaction on Power Apparatus and System, Vol. PAS-94, pp. 827-8833, May/June, 1975

13. DeMello, F.P., Hannet, L.N. and Undrili, J.M., "Practical Approches to Suplementary Stabilizing from accelerating Power", IEEE Transaction on Power Apparatus and System, Vol.PAS-97, pp. 1515-1522, September/October, 1978

14. Bayne, J.P., Lee, D.C. and Watson, W., "Power System Stabilizer for Thermal Units Based on Deviation of Accelerating Power", IEEE Transaction on Power Apparatus and System, Vol. PAS-96, pp. 1777-1783, 1977

15. Elgerd, O.I., "Electric Energy System Theory : An Introduction" McGraw-Hill Book Company, 1971

16. Kimbark, E.W., "Power System Stability" Vol. 1, Wiley, New York, 1948

17. Bauer, D.L., Bhun, W.D., Cogwell, S.S., Ostroski, D.B., and Swanson, D.A., "Simulation of Low Frequency Undamped Oscillations in Large Power System", IEEE Transaction on Power Apparatus and System, Vol. PAS-96, pp. 1777-1783, March/April, 1977

ภาคผนวก ก

ทรานเซียนต์รีแอกแตนซ์ และ ซับทรานเซียนต์รีแอกแตนซ์

ในการวิเคราะห์เกี่ยวกับเครื่องกำเนิดไฟฟ้า ในบางครั้งจำเป็นต้องรู้ความสัมพันธ์ระหว่าง X'_D และ X''_D กับ X_D, X_{MD}, X_{SD}, และ X_{FD} ในภาคผนวก ก นี้ ได้แสดงวิธีหาความสัมพันธ์ดังกล่าว

ในการนิยามค่า ซับทรานเซียนต์รีแอกแตนซ์ ได้สมมุติว่าที่เวลา $t = 0^-$ ฟลักซ์คล้อง λ_{FD} และ λ_{SD} มีค่าเป็นศูนย์ ในทันทีทันใดที่แรงดันสามเฟสแบบสมมาตรได้รับการป้อนเข้าที่วงจรไฟฟ้าที่สเตเตอร์ กล่าวคือที่เวลา $t = 0^+$ ฟลักซ์คล้อง λ_{FD} และ λ_{SD} ยังคงมีค่าเป็นศูนย์ ดังนั้นจากสมการ (4.36) และ สมการ (4.50)

$$I_{FD} = C_{3D} I_D \qquad (ก.1)$$

$$I_{SD} = C_{5D} I_D \qquad (ก.2)$$

แทนค่า I_{FD} จากสมการ (ก.1) และ I_{SD} จากสมการ (ก.2) ลงในสมการ (4.15) ได้ผลดังนี้

$$\begin{aligned} \lambda_D &= X_D I_D - C_{3D} I_D - C_{5D} I_D \\ &= X''_D I_D \end{aligned} \qquad (ก.3)$$

โดยที่ X''_D ในสมการ (ก.3) คือ ซับทรานเซียนต์รีแอกแตนซ์ จากสมการ (ก.3) X''_D มีค่าดังสมการ (ก.4)

$$X''_D = X_D - C_{3D} - C_{5D} \qquad (ก.4)$$

แทนค่า C_{3D} จากสมการ (4.42) และ C_{5D} จากสมการ (4.54) ลงในสมการ (ก.4) ได้ผลดังนี้

$$X''_D = X_D - (X_{MD})^2 (X_{FD} + X_{SD})(X_{MD} X_{FD} + X_{MD} X_{SD} + X_{FD} X_{SD})^{-1} \qquad (ก.5)$$

ในการนิยามทรานเซียนต์รีแอกแตนซ์ ได้สมมุติว่าไม่มีวงจรหน่วง (damping circuit) ในแนวแกนตรงของเครื่องกำเนิดไฟฟ้า และได้สมมุติว่าที่เวลา $t = 0^-$ ฟลักซ์คล้อง λ_{FD} มีค่าเป็นศูนย์ ในทันทีทันใดที่แรงดันได้รับการป้อนเข้าที่วงจรไฟฟ้าที่สเตเตอร์ กล่าวคือที่เวลา $t = 0^+$ λ_{FD} ยังคงมีค่าเป็นศูนย์ ดังนั้นจากสมการ (4.77) จึงได้

$$I_{FD} = [(X_{MD})^2(X_{MD} + X_{FD})^{-1}]I_D \qquad (ก.6)$$

และจากสมการ (4.78) จึงได้

$$\lambda_D = X_D I_D - [(X_{MD})^2(X_{MD} + X_{FD})^{-1}]I_D \qquad (ก.7)$$

โดยที่ X'_D ในสมการ (ก.7) คือทรานเซียนต์รีแอกแตนซ์ จากสมการ (ก.7) จึงได้ว่า X'_D มีค่าดังสมการ (ก.8)

$$X'_D = X_D - (X_{MD})^2(X_{MD} + X_{FD})^{-1} \qquad (ก.8)$$

ดังนั้น X'_D และ X''_D จึงมีความสัมพันธ์กับ X_D, X_{MD}, X_{SD}, และ X_{FD} ตามที่แสดงในตารางที่ ก.1

$$X''_D = X_D - (X_{MD})^2(X_{FD} + X_{SD})(X_{MD}X_{FD} + X_{MD}X_{SD} + X_{FD}X_{SD})^{-1}$$
$$X'_D = X_D - (X_{MD})^2(X_{MD} + X_{FD})^{-1}$$

ตารางที่ ก.1 ความสัมพันธ์ของ X'_D และ X''_D กับ X_D, X_{MD}, X_{SD}, และ X_{FD}

ภาคผนวก ข

ความสัมพันธ์ของรีแอกแตนซ์ซึ่งพิจารณาจากฟลักซ์รั่ว ของวงจรไฟฟ้าในแนวแกนตรงของโรเตอร์ กับ ทรานเซียนต์รีแอกแตนซ์ ซับทรานเซียนต์รีแอกแตนซ์ และ ซิงโครนัสรีแอกแตนซ์

ในการวิเคราะห์เกี่ยวกับเครื่องกำเนิดไฟฟ้า ในบางครั้งจำเป็นต้องรู้ความสัมพันธ์ของ X_{SD} กับ X'_D ,X''_D และ X_D และความสัมพันธ์ของ X_{FD} กับ X'_D ,X''_D และ X_D ในภาคผนวก ข นี้ ได้แสดงวิธีหาความสัมพันธ์ดังกล่าว

จากสมการ (ก.8)

$$X'_D = X_D - (X_{MD})^2(X_{MD} + X_{FD})^{-1} \qquad (ข.1)$$

สมการ (ข.1) สามารถจัดรูปใหม่ได้ดังสมการ (ข.2)

$$X_D - X'_D = (X_{MD})^2(X_{MD} + X_{FD})^{-1} \qquad (ข.2)$$

คูณตลอดสมการ (ข.2) ด้วย $(X_{MD} + X_{FD})(X_D - X'_D)^{-1}$ ได้ผลดังสมการ (ข.3)

$$X_{MD} + X_{FD} = (X_{MD})^2(X_D - X'_D)^{-1} \qquad (ข.3)$$

ลบพจน์ทางซ้ายและขวาของสมการ (ข.3) ด้วย X_{MD} ได้ผลดังสมการ (ข.4)

$$\begin{aligned} X_{FD} &= (X_{MD})^2(X_D - X'_D)^{-1} - X_{MD} \\ &= X_{MD}[X_{MD}(X_D - X'_D)^{-1} - 1] \\ &= X_{MD}(X_{MD} - X_D + X'_D)(X_D - X'_D)^{-1} \end{aligned} \qquad (ข.4)$$

เนื่องจาก

$$X_{MD} = X_D - X_A \qquad (ข.5)$$

แทนค่า X_{MD} จากสมการ (ข.5) ลงในสมการ (ข.4) ได้ผลดังนี้

$$X_{FD} = (X_D - X_A)(X_D - X_A - X_D + \acute{X}_D)(X_D - \acute{X}_D)^{-1}$$
$$= (X_D - X_A)(\acute{X}_D - X_A)(X_D - \acute{X}_D)^{-1} \quad (ข.6)$$

กำหนดให้

$$X_{DV} = X_{MD}X_{FD} + X_{MD}X_{SD} + X_{FD}X_{SD} \quad (ข.7)$$

จากสมการ (ก.5)

$$\acute{\acute{X}}_D = X_D - (X_{MD})^2(X_{FD} + X_{SD})(X_{MD}X_{FD} + X_{MD}X_{SD} + X_{FD}X_{SD})^{-1} \quad (ข.8)$$

จากสมการ (ข.7) และ (ข.8)

$$\acute{\acute{X}}_D = X_D - (X_{MD})^2(X_{FD} + X_{SD})(X_{DV})^{-1} \quad (ข.9)$$

ลบพจน์ทางซ้ายและขวาของสมการ (ข.1) ด้วย X_A ได้ผลดังสมการ (ข.10)

$$\acute{X}_D - X_A = X_D - X_A - (X_{MD})^2(X_{MD} + X_{FD})^{-1} \quad (ข.10)$$

แทนค่า $X_D - X_A$ ในสมการ (ข.10) ด้วย X_{MD} ได้ผลดังสมการ (ข.11)

$$\acute{X}_D - X_A = X_{MD} - (X_{MD})^2(X_{MD} + X_{FD})^{-1}$$
$$= X_{MD}X_{FD}(X_{MD} + X_{FD})^{-1} \quad (ข.11)$$

ลบพจน์ทางซ้ายและขวาของสมการ (ข.9) ด้วย X_A ได้ผลดังสมการ (ข.12)

$$\acute{\acute{X}}_D - X_A = X_D - X_A - (X_{MD})^2(X_{FD} + X_{SD})(X_{DV})^{-1} \quad (ข.12)$$

แทนค่า $X_D - X_A$ ในสมการ (ข.12) ด้วย X_{MD} ได้ผลดังสมการ (ข.13)

$$\begin{aligned} X''_D - X_A &= X_{MD} - (X_{MD})^2(X_{FD} + X_{SD})(X_{DV})^{-1} \\ &= X_{MD}[X_{DV} - X_{MD}(X_{FD} + X_{SD})](X_{DV})^{-1} \end{aligned} \qquad (ข.13)$$

แทนค่า X_{DV} จากสมการ (ข.7) ลงในสมการ (ข.13) ได้ผลดังนี้

$$\begin{aligned} X''_D - X_A &= X_{MD}(X_{MD}X_{FD} + X_{MD}X_{SD} + X_{FD}X_{SD} - X_{MD}X_{FD} - X_{MD}X_{SD})(X_{DV})^{-1} \\ &= X_{MD}X_{FD}X_{SD}(X_{DV})^{-1} \end{aligned} \qquad (ข.14)$$

จากสมการ (ข.1) และ สมการ (ข.9)

$$\begin{aligned} X'_D - X''_D &= (X_{MD})^2(X_{FD} + X_{SD})(X_{DV})^{-1} - (X_{MD})^2(X_{MD} + X_{FD})^{-1} \\ &= (X_{MD})^2[(X_{FD} + X_{SD})(X_{MD} + X_{FD}) - X_{DV}][X_{DV}(X_{MD} + X_{FD})]^{-1} \\ &= (X_{MD})^2[X_{MD}X_{FD} + (X_{FD})^2 + X_{MD}X_{SD} + \\ &\quad X_{FD}X_{SD} - X_{DV}][X_{DV}(X_{MD} + X_{FD})]^{-1} \\ &= (X_{MD})^2[X_{DV} + (X_{FD})^2 - X_{DV}][X_{DV}(X_{MD} + X_{FD})]^{-1} \\ &= (X_{MD}X_{FD})^2[X_{DV}(X_{MD} + X_{FD})]^{-1} \end{aligned} \qquad (ข.15)$$

สมการ (ข.14) หารด้วยสมการ (ข.15) ได้ผลดังนี้

$$(X''_D - X_A)(X'_D - X''_D)^{-1} = X_{SD}(X_{MD} + X_{FD})(X_{MD}X_{FD})^{-1} \qquad (ข.16)$$

จากสมการ (ข.11) และ สมการ (ข.16)

$$(X'_D - X_A)(X''_D - X_A)(X'_D - X''_D)^{-1} = X_{SD} \qquad (ข.17)$$

ดังนั้น

$$X_{SD} = (X'_D - X_A)(X''_D - X_A)(X'_D - X''_D)^{-1} \qquad (ข.18)$$

ภาคผนวก ค

ความสัมพันธ์ระหว่างค่าคงตัวในแบบจำลองเชิงเส้นในแนวแกนตรงของเครื่องกำเนิดไฟฟ้ากับค่าซีบทรานเซียนต์รีแอกแตนซ์ ทรานเซียนต์รีแอกแตนซ์ และรีแอกแตนซ์ซึ่งพิจารณาจากฟลักซ์รั่ว

ในภาคผนวก ค นี้ ได้แสดงให้เห็นถึงความสัมพันธ์ระหว่าง X''_D, X'_D และ X_A กับค่าคงตัวในแบบจำลองเชิงเส้นในแนวแกนตรงของเครื่องกำเนิดไฟฟ้า

กำหนดให้

$$X_D - X_A = X_1 \qquad (ค.1)$$

$$X'_D - X_A = X_2 \qquad (ค.2)$$

$$X''_D - X_A = X_3 \qquad (ค.3)$$

$$X_D - X'_D = X_4 \qquad (ค.4)$$

$$X'_D - X''_D = X_5 \qquad (ค.5)$$

จากสมการที่ (ข.6)

$$X_{FD} = (X_D - X_A)(X'_D - X_A)(X_D - X'_D)^{-1} \qquad (ค.6)$$

แทนค่า X_1 จากสมการ (ค.1) X_2 จากสมการ (ค.2) และ X_4 จากสมการ (ค.4) ลงในสมการ (ค.6) ได้ผลดังสมการ (ค.7)

$$X_{FD} = X_1 X_2 (X_4)^{-1} \qquad (ค.7)$$

จากสมการที่ (ข.18)

$$X_{SD} = (X'_D - X_A)(X''_D - X_A)(X'_D - X''_D)^{-1} \qquad (ค.8)$$

แทนค่า X_2 จากสมการ (ค.2) X_3 จากสมการ (ค.3) และ X_5 จากสมการ (ค.5) ลงในสมการ (ค.8) ได้ผลดังนี้

$$X_{SD} = X_2X_3(X_5)^{-1} \quad (ค.9)$$

เนื่องจาก

$$X_{MD} = X_D - X_A \quad (ค.10)$$

ดังนั้น จากสมการ (ค.1)

$$X_{MD} = X_1 \quad (ค.11)$$

จากสมการ (ค.11) และสมการ (ค.7)

$$X_{MD}X_{FD} = (X_1)^2X_2(X_4)^{-1} \quad (ค.12)$$

จากสมการ (ค.11) และจากสมการ (ค.9)

$$X_{MD}X_{SD} = X_1X_2X_3(X_5)^{-1} \quad (ค.13)$$

จากสมการ (ค.7) และจากสมการ (ค.9)

$$X_{FD}X_{SD} = X_1(X_2)^2X_3(X_4X_5)^{-1} \quad (ค.14)$$

กำหนดให้

$$X_{DV} = X_{MD}X_{FD} + X_{MD}X_{SD} + X_{FD}X_{SD} \quad (ค.15)$$

แทนค่า $X_{MD}X_{FD}$, $X_{MD}X_{SD}$ และ $X_{FD}X_{SD}$ จากสมการ (ค.12), (ค.13) และ (ค.14) ตามลำดับ ลงในสมการ (ค.15) ได้ผลดังสมการ (ค.16)

$$\begin{aligned} X_{DV} &= [(X_1)^2X_2(X_4)^{-1}] + [X_1X_2X_3(X_5)^{-1}] + [X_1(X_2)^2X_3(X_4X_5)^{-1}] \\ &= X_1X_2[X_1(X_4)^{-1} + X_3(X_5)^{-1} + X_2X_3(X_4X_5)^{-1} \\ &= X_1X_2(X_1X_5 + X_3X_4 + X_2X_3)(X_4X_5)^{-1} \end{aligned} \quad (ค.16)$$

กำหนดให้

$$X_6 = X_D X''_D \quad (ค.17)$$

$$X_7 = X'_D X_A \quad (ค.18)$$

$$X_8 = X''_D X_A \quad (ค.19)$$

$$X_9 = X'_D X''_D \quad (ค.20)$$

จากสมการ (ค.1) และสมการ (ค.5)

$$\begin{aligned} X_1X_5 &= (X_D - X_A)(X'_D - X''_D) \\ &= X_DX'_D - X_DX''_D - X'_DX_A + X''_DX_A \\ &= X_DX'_D - X_6 - X_7 + X_8 \end{aligned} \quad (ค.21)$$

จากสมการ (ค.3) และสมการ (ค.4)

$$\begin{aligned} X_3X_4 &= (X''_D - X_A)(X_D - X'_D) \\ &= X_DX''_D - X'_DX''_D - X_DX_A + X'_DX_A \\ &= X_6 - X_9 - X_DX_A + X_7 \end{aligned} \quad (ค.22)$$

จากสมการ (ค.2) และสมการ (ค.3)

$$\begin{aligned} X_2X_3 &= (X'_D - X_A)(X''_D - X_A) \\ &= X'_DX''_D - X'_DX_A - X''_DX_A + (X_A)^2 \\ &= X_9 - X'_DX_A - X_8 + (X_A)^2 \end{aligned} \quad (ค.23)$$

จากสมการ (ค.21) (ค.22) และ (ค.23)

$$\begin{aligned} X_1X_5 + X_3X_4 + X_2X_3 &= X_DX'_D - X_DX_A - X'_DX_A + (X_A)^2 \\ &= X_D(X'_D - X_A) - X_A(X'_D - X_A) \\ &= (X_D - X_A)(X'_D - X_A) \end{aligned} \quad (ค.24)$$

จากสมการ (ค.1) (ค.2) และ (ค.24)

$$X_1X_5 + X_3X_4 + X_2X_3 = X_1X_2 \qquad (ค.25)$$

แทนค่า $X_1X_5 + X_3X_4 + X_2X_3$ จากสมการ (ค.25) ลงในสมการ (ค.16) ได้ผลดังสมการ (ค.26)

$$X_{DV} = (X_1X_2)^2(X_4X_5)^{-1} \qquad (ค.26)$$

จากตารางที่ 4.4

$$C_{1D} = (X_{MD} + X_{FD})(X_{MD} + X_{SD})(X_{DV})^{-1}$$
$$= [(X_{MD})^2 + X_{MD}X_{FD} + X_{MD}X_{SD} + X_{FD}X_{SD}](X_{DV})^{-1} \qquad (ค.27)$$

แทนค่า X_{MD} จากสมการ (ค.11) และ $X_{MD}X_{FD} + X_{MD}X_{SD} + X_{FD}X_{SD}$ จากสมการ (ค.15) ลงในสมการ (ค.27) ได้ผลดังสมการ (ค.28)

$$C_{1D} = [(X_1)^2 + X_{DV}](X_{DV})^{-1}$$
$$= 1 + (X_1)^2(X_{DV})^{-1} \qquad (ค.28)$$

แทนค่า X_{DV} จากสมการ (ค.26) ลงในสมการ (ค.28) ได้สมการ (ค.29)

$$C_{1D} = 1 + (X_4X_5)(X_2)^{-2} \qquad (ค.29)$$

แทนค่า X_2 จากสมการ (ค.2) X_4 จากสมการ (ค.4) และ X_5 จากสมการ (ค.5) ลงในสมการ (ค.29) ได้ผลดังสมการ (ค.30)

$$C_{1D} = 1 + (X_D - X'_D)(X'_D - X''_D)(X'_D - X_A)^{-2} \qquad (ค.30)$$

จากตารางที่ 4.4

$$C_{2D} = X_{MD}(X_{MD} + X_{SD})(X_{DV})^{-1} \qquad (ค.31)$$

แทนค่า X_{MD} จากสมการ (ค.11) และแทนค่า X_{SD} จากสมการ (ค.9) ลงใน

สมการ (ค.31) ได้ผลดังนี้

$$C_{2D} = X_1[X_1 + X_2X_3(X_5)^{-1}](X_{DV})^{-1}$$
$$= X_1[X_1X_5 + X_2X_3](X_5X_{DV})^{-1} \quad (ค.32)$$

แทนค่า X_{DV} จากสมการ (ค.26) ลงในสมการ (ค.32) ได้ผลดังนี้

$$C_{2D} = X_1(X_5)^{-1}(X_1X_5 + X_2X_3)X_4X_5(X_1X_2)^{-2}$$
$$= X_4(X_2)^{-1}(X_1X_5 + X_2X_3)(X_1X_2)^{-1}$$
$$= X_4(X_2)^{-1}[X_5(X_2)^{-1} + X_3(X_1)^{-1}] \quad (ค.33)$$

แทนค่า X_2 จากสมการ (ค.2) X_3 จากสมการ (ค.3) X_4 จากสมการ (ค.4) และ X_5 จากสมการ (ค.5) ลงในสมการ (ค.33) ได้ผลดังสมการ (ค.34)

$$C_{2D} = (X_D - X'_D)(X'_D - X_A)^{-1}[(X'_D - X''_D)(X'_D - X_A)^{-1} +$$
$$(X''_D - X_A)(X_D - X_A)^{-1}] \quad (ค.34)$$

จากตารางที่ 4.4

$$C_{3D} = (X_{MD})^2X_{SD}(X_{DV})^{-1} \quad (ค.35)$$

แทนค่า X_{MD} จากสมการ (ค.11) และ แทนค่า X_{SD} จากสมการ (ค.9) ลงในสมการ ค.35) ได้ผลดังนี้

$$C_{3D} = (X_1)^2X_2X_3(X_5X_{DV})^{-1} \quad (ค.36)$$

แทนค่า X_{DV} จากสมการ (ค.26) ลงในสมการ (ค.36) ได้ผลดังสมการ (ค.37)

$$C_{3D} = (X_1)^2X_2X_3X_4X_5(X_1X_2)^{-2}(X_5)^{-1}$$
$$= X_3X_4(X_2)^{-1} \quad (ค.37)$$

แทนค่า X_2 จากสมการ (ค.2) X_3 จากสมการ (ค.3) และ X_4 จากสมการ

(ค.4) ลงในสมการ (ค.37) ได้ผลดังสมการ (ค.38)

$$C_{3D} = (X''_D - X_A)(X_D - X'_D)(X'_D - X_A)^{-1} \qquad (ค.38)$$

จากตารางที่ 4.4

$$C_{4D} = X_{MD}(X_{MD} + X_{FD})(X_{DV})^{-1} \qquad (ค.39)$$

แทนค่า X_{MD} จากสมการ (ค.11) และแทนค่า X_{FD} จากสมการ (ค.7) ลงในสมการ (ค.39) ได้ผลดังนี้

$$\begin{aligned} C_{4D} &= X_1[X_1 + X_1X_2(X_4)^{-1}](X_{DV})^{-1} \\ &= (X_1)^2[1 + X_2(X_4)^{-1}](X_{DV})^{-1} \\ &= (X_1)^2(X_{DV}X_4)^{-1}(X_2 + X_4) \end{aligned} \qquad (ค.40)$$

บวกสมการ (ค.2) กับสมการ (ค.4) เข้าด้วยกัน ได้ผลดังนี้

$$\begin{aligned} X_2 + X_4 &= (X'_D - X_A) + (X_D - X'_D) \\ &= X_D - X_A \end{aligned} \qquad (ค.41)$$

แทนค่า $X_2 + X_4$ จากสมการ (ค.41) ลงในสมการ (ค.40) ได้ผลดังนี้

$$C_{4D} = (X_D - X_A)(X_1)^2(X_{DV}X_4)^{-1} \qquad (ค.42)$$

แทนค่า X_{DV} จากสมการ (ค.26) ลงในสมการ (ค.42) ได้ผลดังนี้

$$\begin{aligned} C_{4D} &= (X_D - X_A)(X_1)^2X_4X_5(X_1X_2)^{-2}(X_4)^{-1} \\ &= (X_D - X_A)X_1X_5(X_2)^{-2} \end{aligned} \qquad (ค.43)$$

แทนค่า X_2 จากสมการ (ค.2) และ X_3 จากสมการ (ค.3) ลงในสมการ (ค.43) ได้ผลดังนี้

$$C_{4D} = (X_D - X_A)(\dot{X}_D - \ddot{X}_D)(\dot{X}_D - X_A)^{-2} \quad \text{(ค.44)}$$

จากตารางที่ 4.4

$$C_{5D} = (X_{MD})^2 X_{FD} (X_{DV})^{-1} \quad \text{(ค.45)}$$

แทนค่า X_{MD} จากสมการ (ค.11) และ แทนค่า X_{FD} จากสมการ (ค.7) ลงในสมการ (ค.45) ได้ผลดังสมการ (ค.46)

$$C_{5D} = (X_1)^3 X_2 (X_{DV} X_4)^{-1} \quad \text{(ค.46)}$$

แทนค่า X_{DV} จากสมการ (ค.26) ลงในสมการ (ค.46) ได้ผลดังนี้

$$\begin{aligned} C_{5D} &= (X_1)^3 X_2 X_4 X_5 (X_1 X_2)^{-2} (X_4)^{-1} \\ &= X_1 X_5 (X_2)^{-1} \end{aligned} \quad \text{(ค.47)}$$

แทนค่า X_1 จากสมการ (ค.1) X_2 จากสมการ (ค.2) และ X_5 จากสมการ (ค.5) ลงในสมการ (ค.47) ได้ผลดังนี้

$$C_{5D} = (X_D - X_A)(\dot{X}_D - \ddot{X}_D)(\dot{X}_D - X_A)^{-1} \quad \text{(ค.48)}$$

จากตารางที่ 4.4

$$\begin{aligned} C_{6D} &= X_{SD}(X_{MD} + X_{FD})(X_{DV})^{-1} \\ &= (X_{MD}X_{SD} + X_{FD}X_{SD})(X_{DV})^{-1} \\ &= X_{MD}X_{SD}(X_{DV})^{-1} + X_{FD}X_{SD}(X_{DV})^{-1} \end{aligned} \quad \text{(ค.49)}$$

แทนค่า $X_{MD}X_{SD}$ จากสมการ (ค.13) และ แทนค่า $X_{FD}X_{SD}$ จากสมการ (ค.14) ลงในสมการ (ค.49) ได้ผลดังนี้

$$\begin{aligned} C_{6D} = {} & [X_1 X_2 X_3 (X_5)^{-1}][X_4 X_5 (X_1 X_2)^{-2}] + \\ & [X_1 (X_2)^2 X_3 (X_4 X_5)^{-1}][X_4 X_5 (X_1 X_2)^{-2}] \end{aligned}$$

$$= X_3X_4(X_1X_2)^{-1} + X_2X_3(X_1X_2)^{-1}$$
$$= X_3(X_4+ X_2)(X_1X_2)^{-1} \qquad (ค.50)$$

แทนค่า X_2 จากสมการ (ค.2) และ X_4 จากสมการ (ค.4) ลงในสมการ (ค.50) ได้ผลดังสมการ (ค.51)

$$C_{6D} = X_3(X_D- \acute{X}_D+ \acute{X}_D- X_A)(X_1X_2)^{-1}$$
$$= X_3(X_D- X_A)(X_1X_2)^{-1} \qquad (ค.51)$$

แทนค่า X_1 จากสมการ (ค.1) X_2 จากสมการ (ค.2) และ X_3 จากสมการ (ค.3) ลงในสมการ (ค.51) ได้ผลดังนี้

$$C_{6D} = (\acute{\acute{X}}_D- X_A)(X_D- X_A)(X_D- X_A)^{-1}(\acute{X}_D- X_A)^{-1}$$
$$= (\acute{\acute{X}}_D- X_A)(\acute{X}_D- X_A)^{-1} \qquad (ค.52)$$

จากตารางที่ 4.4

$$C_{7D} = X_{FD}(X_{MD}+ X_{SD})(X_{DV})^{-1}$$
$$= X_{FD}X_{MD}(X_{DV})^{-1} + X_{FD}X_{SD}(X_{DV})^{-1} \qquad (ค.53)$$

แทนค่า $X_{MD}X_{FD}$ จากสมการ (ค.12) และ แทนค่า $X_{MD}X_{SD}$ จากสมการ (ค.14) ลงในสมการ (ค.53) ได้ผลดังนี้

$$C_{7D} = (X_1)^2X_2(X_4)^{-1}X_4X_5(X_1X_2)^{-2} +$$
$$X_1(X_2)^2X_3(X_4X_5)^{-1}X_4X_5(X_1X_2)^{-2}$$
$$= X_5(X_2)^{-1} + X_3(X_1)^{-1} \qquad (ค.54)$$

แทนค่า X_1 จากสมการ (ค.1) X_2 จากสมการ (ค.2) X_3 จากสมการ (ค.3) และ X_5 จากสมการ (ค.5) ลงในสมการ (ค.54) ได้ผลดังนี้

$$C_{7D} = (\acute{X}_D- \acute{\acute{X}}_D)(\acute{X}_D- X_A) + (\acute{\acute{X}}_D- X_A)(X_D- X_A)^{-1} \qquad (ค.55)$$

ดังนั้นค่าคงตัวในแบบจำลองเชิงเส้นชนิดแนวแกนตรงของเครื่องกำเนิดไฟฟ้า ที่อยู่ในรูปฟังก์ชันของ X_D, $\acute{X}_D$, $\acute{\acute{X}}_D$ และ X_A จึงมีค่าตามที่แสดงในตารางที่ ค.1

$$C_{1D} = 1 + (X_D - \acute{X}_D)(\acute{X}_D - \acute{\acute{X}}_D)(\acute{X}_D - X_A)^{-2}$$

$$C_{2D} = (X_D - \acute{X}_D)(\acute{X}_D - X_A)^{-1}[(\acute{X}_D - \acute{\acute{X}}_D)(\acute{X}_D - X_A)^{-1} + (\acute{\acute{X}}_D - X_A)(X_D - X_A)^{-1}]$$

$$C_{3D} = (\acute{\acute{X}}_D - X_A)(X_D - \acute{X}_D)(\acute{X}_D - X_A)^{-1}$$

$$C_{4D} = (X_D - X_A)(\acute{X}_D - \acute{\acute{X}}_D)(\acute{X}_D - X_A)^{-2}$$

$$C_{5D} = (X_D - X_A)(\acute{X}_D - \acute{\acute{X}}_D)(\acute{X}_D - X_A)^{-1}$$

$$C_{6D} = (\acute{\acute{X}}_D - X_A)(\acute{X}_D - X_A)^{-1}$$

$$C_{7D} = (\acute{X}_D - \acute{\acute{X}}_D)(\acute{X}_D - X_A) + (\acute{\acute{X}}_D - X_A)(X_D - X_A)^{-1}$$

ตารางที่ ค.1 ค่าคงตัวในแบบจำลองเชิงเส้นในแนวแกนตรงของเครื่องกำเนิดไฟฟ้า ซึ่งอยู่ในรูปฟังก์ชันของ ซับทรานเซียนต์รีแอกแตนซ์ ทรานเซียนต์รีแอกแตนซ์ และรีแอกแตนซ์ซิ่งพิจารณาจากฟลักซ์รั่ว

ภาคผนวก ง

การใช้โปรแกรมสำหรับวิเคราะห์ออสซิลเลชันความถี่ต่ำ ที่เกิดขึ้นเองในระบบไฟฟ้ากำลัง

โปรแกรมสำหรับวิเคราะห์ออสซิลเลชันความถี่ต่ำที่เกิดขึ้นเองในระบบไฟฟ้ากำลัง ที่ได้พัฒนาขึ้นนี้ เป็นโปรแกรมที่ใช้หาค่าของค่าไอเก็นที่สัมพันธ์กับออสซิลเลชันของโรเตอร์ ของเครื่องกำเนิดไฟฟ้าในระบบไฟฟ้ากำลัง โปรแกรมดังกล่าวมีชื่อว่า LFO โปรแกรม LFO นี้ได้รับการพัฒนาขึ้นมาให้สามารถใช้กับไมโครคอมพิวเตอร์ (microcomputer) ชนิด 16 บิท ตระกูล IBM PC/XT โดยต้องทำงานภายใต้ MS-DOS

ง.1 การเข้าสู่โปรแกรม LFO

เมื่อเปิดเครื่องคอมพิวเตอร์แล้วทำการโหลด MS-DOS จนเรียบร้อย ที่จอภาพจะปรากฏเครื่องหมาย A > การเข้าสู่โปรแกรม LFO ทำได้โดยพิมพ์ LFO ต่อจาก A > แล้วกด Return เมื่อโปรแกรมฯเริ่มทำงาน จะมีข้อความตามที่ได้แสดงในกรอบข้างล่างปรากฏขึ้นบนจอภาพ

```
    CALCULATION SELECTION MODE
(0) EIGENVALUE SEARCH
(1) SWEEP FREQUENCY METHOD
    ENTER YOUR SELECTED MODE >
```

เมื่อผู้ใช้พิมพ์ 0 แล้ว กด Return โปรแกรมฯ จะทำงานในโมดของการหาค่าของค่าไอเก็น แต่ถ้าพิมพ์ 1 แล้วกด Return โปรแกรมฯ จะทำงานในโมดของการหาค่าของค่าไอเก็นอย่างหยาบ ๆ ตลอดช่วงของความถี่ที่ผู้ใช้ต้องการ

เมื่อผู้ใช้เลือกโมดการทำงานของโปรแกรมฯ ตามวิธีการข้างต้นแล้ว ข้อความตามที่

ได้แสดงในกรอบข้างล่างจะปรากฏขึ้นบนจอภาพ

```
     PROGRAM SELECTION MODE
(0) NORMAL MODE (READ DATA FORM OPENNED FILE)
(1) INTERACTIVE MODE (READ DATA FORM KEYBOARD)
     ENTER YOUR SELECTION MODE >
```

ผู้ใช้โปรแกรมต้องพิมพ์เลข 0 หรือ 1 ในตำแหน่งของ เคอร์เซอร์ (cursor) ต่อจากข้อความ ENTER YOUR SELECTION MODE > การพิมพ์ 0 หมายความว่าผู้ใช้ได้จัดไฟล์ข้อมูลไว้เรียบร้อยแล้ว (รายละเอียดของการจัดไฟล์ข้อมูลจะได้อธิบายภายหลัง) ส่วนการพิมพ์ 1 หมายความว่าผู้ใช้ต้องการป้อนข้อมูลโดยผ่านทางแป้นพิมพ์

ง.2 การป้อนข้อมูลผ่านทางแป้นพิมพ์

ในกรณีที่ผู้ใช้ต้องการป้อนข้อมูลผ่านทางแป้นพิมพ์ ภายหลังจากเลือกโมดการป้อนข้อมูลโดยพิมพ์เลข 1 ต่อจากข้อความ ENTER YOUR SELECTION MODE > แล้วกด Return ข้อความภาษาอังกฤษตามที่ได้แสดงในกรอบข้างล่างจะปรากฏขึ้นมาทีละบรรทัด ข้อความในวงเล็บต่อจากข้อความภาษาอังกฤษคือข้อมูลที่ผู้ใช้ต้องพิมพ์เข้าไป

```
TOTAL NUMBER OF BUSES = (จำนวนบัสของระบบไฟฟ้ากำลัง)
TOTAL NUMBER OF LINES = (จำนวนสายส่งของระบบไฟฟ้ากำลัง)
TOTAL NUMBER OF TRANSFORMERS = (จำนวนหม้อแปลง)
TOTAL NUMBER OF GENERATORS = (จำนวนเครื่องกำเนิดไฟฟ้า)
FORCED SYNCHRONOUS MACHINE IS GEN. NO. = (เครื่องกำเนิดไฟฟ้า
ที่ถูกบังคับ)
FREQUENCY OF POWER SYSTEM = (ความถี่ระบุ ของระบบไฟฟ้ากำลัง)
BASE MVA OF SYSTEM = (ค่าเบสของ MVA ของระบบไฟฟ้ากำลัง)
```

```
PER UNIT SPEED DEVIATION OF FORCED GEN. = (ความเร็วเบี่ยงเบน
ของเครื่องกำเนิดไฟฟ้าที่ถูกบังคับ)
MAXIMUM ITERATIONS FOR EIGENVALUE SEARCH = (จำนวนวงรอบของ
การค้นหาค่าของค่าไอเก็นที่มากที่สุดที่ผู้ใช้ยินยอม)
TOLERANCE FOR EIGENVALUE SEARCH = (ความคลาดเคลื่อนของค่าไอเก็น
ที่ผู้ใช้ยอมได้)
INITIAL INCREMENTAL FREQUENCY IN HERTZ = (ส่วนจินตภาพของค่า
ไอเก็นเริ่มต้น
LOWEST FREQUENCY OF FREQUENCY BAND IN HERTZ = (ความถี่เริ่มต้น
ตามวิธีการ sweep frequency )
FREQUENCY INCREMENT IN HERTZ = (ความถี่ที่เพิ่มในแต่ละวงรอบของ
การคำนวนตามวิธีการ sweep frequency )
MAXIMUM POINTS OF FREQUENCY REQUIRED = (จำนวนจุดในแบนด์ของ
ความถี่ ของการคำนวณตามวิธีการ sweep frequency )
```

เมื่อผู้ใช้ป้อนข้อมูลข้างต้นเรียบร้อยแล้ว ลำดับต่อมาข้อความที่เกี่ยวข้องกับข้อมูลของสายส่ง ตามที่ได้แสดงในกรอบข้างล่าง จะปรากฏขึ้นมาทีละบรรทัด โดยเริ่มต้นจากข้อมูลที่เกี่ยวข้องกับสายส่งหมายเลข 1 ผู้ใช้ต้องพิมพ์ตัวเลขข้อมูลต่าง ๆ ของสายส่งจนครบทุกสาย

```
LINE NO. 1 STARTING FORM BUS NO. (บัสที่ต้นทางของสายส่ง)
LINE NO. 1 TERMINATED ON BUS NO. (บัสที่ปลายทางของสายส่ง)
RESISTANCE IN PU. OF LINE NO. 1  = (ความต้านของสายส่งในระบบ
เปอร์ยูนิต)
REACTANCE IN PU. OF LINE NO. 1  = (รีแอกแตนซ์ของสายส่งในระบบ
เปอร์ยูนิต)
CHARGING MVA OF LINE NO. 1  =  (Charging MVA ของสายส่ง)
```

เมื่อป้อนข้อมูลของสายส่ง จนครบทุกสาย ลำดับต่อมาข้อความที่เกี่ยวข้องกับข้อมูล

ของหม้อแปลง ตามที่ได้แสดงในกรอบข้างล่าง จะปรากฏขึ้นมาทีละบรรทัด โดยเริ่มต้นจากข้อมูลที่เกี่ยวข้องกับหม้อแปลงหมายเลข 1 ผู้ใช้ต้องพิมพ์ตัวเลขข้อมูลต่าง ๆ ของหม้อแปลงจนครบทุกหม้อแปลง

PRIMARY SIDE OF TRANSFORMER NO. 1 = (บัสทางด้านปฐมภูมิของหม้อแปลง)
SECONDARY SIDE OF TRANSFORMER NO. 1 = (บัสทางด้านทุติยภูมิของหม้อแปลง)
RESISTANCE IN PU. OF TRANSFORMER NO. 1 = (ความต้านทานของหม้อแปลง)
REACTANCE IN PU. OF TRANSFORMER NO. 1 = (รีแอกแตนซ์ของหม้อแปลง)
TAP SETTING OF TRANSFORMER NO. 1 = (แท็ปของหม้อแปลง)
PHASE SHIFT IN DEGRE OF TRANSFORMER NO. 1 = (จำนวนองศาของการเลื่อนของเฟสของหม้อแปลง ในกรณีที่หม้อแปลงเป็นหม้อแปลงเลื่อนเฟส)

ลำดับต่อมาข้อความที่เกี่ยวข้องกับข้อมูลของบัส ตามที่ได้แสดงในกรอบข้างล่าง จะปรากฏขึ้นมาทีละบรรทัด โดยเริ่มต้นจากข้อมูลที่เกี่ยวข้องกับบัสหมายเลข 1 ผู้ใช้ต้องพิมพ์ตัวเลขข้อมูลต่าง ๆ ของบัสจนครบทุกบัส

REAL POWER GENERATION IN MVA AT BUS NO. 1 = (กำลังไฟฟ้าจริงที่ผลิต)
REACTIVE POWER GENERATION IN MVA AT BUS NO. 1 = (กำลังไฟฟ้ารีแอกตีฟที่ผลิต)
REAL POWER LOAD IN MVA AT BUS NO. 1 = (กำลังไฟฟ้าจริงของโหลด)
REACTIVE POWER LOAD IN MVA AT BUS NO. 1 = (กำลังไฟฟ้ารีแอกตีฟของโหลด)
MAGNITUDE OF VOLTAGE IN PU. AT BUS NO. 1 = (ขนาดของแรงดัน)
PHASE ANGLE OF VOLTAGE IN DEGRE AT BUS NO. 1 = (เฟสของแรงดัน)
FIXED CAPACITANCE IN PU. AT BUS NO. 1 = (ค่าความจุไฟฟ้าในระบบเปอร์ยูนิตของตัวเก็บประจุที่ต่ออยู่ที่บัสนั้น ๆ)
TYPE OF BUS NO. 1 = (ชนิดของบัส พิมพ์ 1 สำหรับบัสของเครื่องกำเนิดไฟฟ้า พิมพ์ 2 สำหรับบัสของโหลด พิมพ์ 3 สำหรับ อินฟินิตบัส)

ลำดับต่อมาข้อความที่เกี่ยวข้องกับข้อมูลของเครื่องกำเนิดไฟฟ้า ตามที่ได้แสดงในกรอบข้างล่าง จะปรากฏขึ้นมาทีละบรรทัด โดยเริ่มต้นจากข้อมูลที่เกี่ยวข้องกับเครื่องกำเนิดไฟฟ้าหมายเลข 1 ผู้ใช้ต้องพิมพ์ตัวเลขข้อมูลต่าง ๆ ของเครื่องกำเนิดไฟฟ้าจนครบทุกเครื่อง

D - AXIS SYNCHRONOUS REACTANCE OF GEN. NO. 1 = (X_D)

D - AXIS TRANSIENT REACTANCE OF GEN. NO. 1 = (X'_D)

D - AXIS SUBTRANSIENT REACTANCE OF GEN. NO. 1 = (X''_D)

Q - AXIS SYNCHRONOUS REACTANCE OF GEN. NO. 1 = (X_Q)

Q - AXIS TRANSIENT REACTANCE OF GEN. NO. 1 = (X'_Q)

Q - AXIS SUBTRANSIENT REACTANCE OF GEN. NO. 1 = (X''_Q)

LEAKAGE REACTANCE OF GEN. NO. 1 = (X_L)

ARMATURE RESISTANCE OF GEN. NO. 1 = (R_A)

POINTER REACTANCE OF GEN. NO. 1 = (X_P)

SATURATION FUNCTION AT 1.0 PU. VOLTAGE OF GEN. NO. 1 = (ค่าของฟังก์ชันของการอิ่มตัวที่ระดับแรงดัน 1.0 เปอร์ยูนิต)

SATURATION FUNCTION AT 1.2 PU. VOLTAGE OF GEN. NO. 1 = (ค่าของฟังก์ชันของการอิ่มตัวที่ระดับแรงดัน 1.2 เปอร์ยูนิต)

INERTIA CONSTANT OF GEN. NO. 1 = (ค่าคงตัวแห่งความเฉื่อย H)

MVA RATING OF GEN. NO. 1 = (พิกัด MVA)

DAMPING COEFFICIENT OF GEN. NO. 1 = (สัมประสิทธิ์ของการหน่วง D)

D - AXIS OPENNED CIRCUIT TRANSIENTTIME CONSTANT OF GEN. NO. 1 = (T'_{DO})

D - AXIS OPENNED CIRCUIT SUBTRANSIENTTIME CONSTANT OF GEN. NO. 1 = (T''_{DO})

Q - AXIS OPENNED CIRCUIT TRANSIENTTIME CONSTANT OF GEN. NO. 1 = (T'_{QO})

Q - AXIS OPENNED CIRCUIT SUBTRANSIENT TIME CONSTANT OF GEN. NO. (T''_{QO})

GEN. NO. 1 CONNECTED TO BUS NO. = (บัสที่ขั้ว)

EXCITER AT GEN. NO. 1 TYPE = (ระหัสของระบบเอกไซเตชัน ดูตารางที่ 5.1

ถ้าไม่มีระบบเอกไซเตชัน พิมพ์ 0)
IF THERE IS A POWER SYSTEM STABILIZER AT GEN. NO. 1 PLEASE KEY "1" (ถ้ามีอุปกรณ์สเตบิไลเซอร์ ให้พิมพ์ 1 ถ้าไม่มี พิมพ์ 0)
IF THERE IS A SPEED-GOVERNING SYSTEM AT GEN. NO. 1 PLEASE KEY "1" (ถ้ามีระบบควบคุมความเร็ว พิมพ์ 1 ถ้าไม่มีให้พิมพ์ 0)

ลำดับต่อมาข้อความที่เกี่ยวข้องกับข้อมูลของระบบเอกไซเตชัน ตามที่ได้แสดงในกรอบข้างล่างจะปรากฏขึ้นมาทีละบรรทัด โดยเริ่มต้นจากข้อมูลที่เกี่ยวข้องกับระบบเอกไซเตชันของเครื่องกำเนิดไฟฟ้าหมายเลข 1 ผู้ใช้ต้องพิมพ์ตัวเลขข้อมูลต่าง ๆ ของระบบเอกไซเตชันของเครื่องกำเนิดไฟฟ้าแต่ละเครื่องจนครบทุกเครื่อง (ดูรูปที่ 5.1 ประกอบ)

VRMAX OF EXCITER NO. 1 = (V_{RMAX})
VRMIN OF EXCITER NO. 1 = (V_{RMIN})
KA OF EXCITER NO. 1 = (K_A)
TA OF EXCITER NO. 1 = (T_A)
TB OF EXCITER NO. 1 = (T_B)
TC OF EXCITER NO. 1 = (T_C)
KE OF EXCITER NO. 1 = (K_E)
TE OF EXCITER NO. 1 = (T_E)
KF OF EXCITER NO. 1 = (K_F)
TF OF EXCITER NO. 1 = (T_F)
SATURATON FUNCTION AT 1.0 PU. OF EXCITER NO. 1 = (ค่าฟังก์ชันการอิ่มตัวของเอกไซเตอร์ที่ระดับแรงดัน 1.0 เปอร์ยูนิต ถ้าไม่คำนึงถึงผลของการอิ่มตัวดังกล่าว ให้พิมพ์ค่า 0.0)
SATURATON FUNCTION AT 1.2 PU. OF EXCITER NO. 1 = (ค่าฟังก์ชันการอิ่มตัวของเอกไซเตอร์ที่ระดับแรงดัน 1.0 เปอร์ยูนิต ถ้าไม่คำนึงถึงผลของการอิ่มตัวดังกล่าว ให้พิมพ์ค่า 0.0)
COMPENSATION RESISTANCE OF EXCITER NO. 1 = (R_C ดูรูปที่ 5.2 โดยที่

ในรูปดังกล่าว $Z_c = R_c + jX_c$)
COMPENSATION REACTANCE OF EXCITER NO. 1 = (R_c ดูรูปที่ 5.2 โดยที่
ในรูปดังกล่าว $Z_c = R_c + jX_c$)
TR OF EXCITER NO. 1 = (T_R ในรูปที่ 5.2)
UPPER LIMIT OF EFD OF EXCITER NO. 1 = (E_{FDMAX})
LOWER LIMIT OF EFD OF EXCITER NO. 1 = (E_{FDMIN})

ลำดับต่อมาข้อความที่เกี่ยวข้องกับข้อมูลของอุปกรณ์สเตบิไลเซอร์ ตามที่ได้แสดงในกรอบข้างล่าง จะปรากฏขึ้นมาทีละบรรทัด โดยเริ่มต้นจากข้อมูลสำหรับอุปกรณ์สเตบิไลเซอร์ของเครื่องกำเนิดไฟฟ้าหมายเลข 1 ผู้ใช้ต้องพิมพ์ตัวเลขข้อมูลต่าง ๆ ของอุปกรณ์ดังกล่าวของเครื่องกำเนิดไฟฟ้าแต่ละเครื่องจนครบทุกเครื่อง (ดูรูปที่ 5.9 ประกอบ)

K1 OF POWER SYSTEM STABILIZER NO. 1 = (K_1)
T1 OF POWER SYSTEM STABILIZER NO. 1 = (T_1)
T3 OF POWER SYSTEM STABILIZER NO. 1 = (T_3)
T4 OF POWER SYSTEM STABILIZER NO. 1 = (T_4)
T5 OF POWER SYSTEM STABILIZER NO. 1 = (T_5)
T6 OF POWER SYSTEM STABILIZER NO. 1 = (T_6)
T7 OF POWER SYSTEM STABILIZER NO. 1 = (T_7)
T8 OF POWER SYSTEM STABILIZER NO. 1 = (T_8)
L1 OF POWER SYSTEM STABILIZER NO. 1 = (L_1)
L2 OF POWER SYSTEM STABILIZER NO. 1 = (L_2)

ลำดับต่อมาข้อความที่เกี่ยวข้องกับข้อมูลของระบบควบคุมความเร็ว ตามที่ได้แสดงในกรอบข้างล่าง จะปรากฏขึ้นมาทีละบรรทัด โดยเริ่มต้นจากข้อมูลที่เกี่ยวข้องกับระบบควบคุมความเร็วของเครื่องกำเนิดไฟฟ้าหมายเลข 1 ผู้ใช้ต้องพิมพ์ตัวเลขข้อมูลต่าง ๆ ของอุปกรณ์ดังกล่าวของเครื่องกำเนิดไฟฟ้าแต่ละเครื่องจนครบทุกเครื่อง (ดูรูปที่ 5.10 ประกอบ)

K1 OF SPEED GOVERNER NO. 1 = (K_1)

K2 OF SPEED GOVERNER NO. 1 = (K_2)

K3 OF SPEED GOVERNER NO. 1 = (K_3)

K4 OF SPEED GOVERNER NO. 1 = (K_4)

T1 OF SPEED GOVERNER NO. 1 = (T_1)

T2 OF SPEED GOVERNER NO. 1 = (T_2)

T3 OF SPEED GOVERNER NO. 1 = (T_3)

T4 OF SPEED GOVERNER NO. 1 = (T_4)

T5 OF SPEED GOVERNER NO. 1 = (T_5)

T6 OF SPEED GOVERNER NO. 1 = (T_6)

ง.3 การพิมพ์ข้อมูลลงในไฟล์ข้อมูล

การป้อนข้อมูลให้กับโปรแกรม LFO สามารถทำได้อีกวิธีคือพิมพ์ข้อมูลลงในไฟล์ข้อมูล ซึ่งมีชื่อว่า LFO.DAT ผู้ใช้จำเป็นต้องรู้วิธีการจัดฟอร์แมตของข้อมูลต่างๆ แต่ละบรรทัดในไฟล์ข้อมูลเปรียบเสมือนกับการ์ด (card) ข้อมูลแต่ละแผ่นของคอมพิวเตอร์แบบดั้งเดิมที่รับข้อมูลผ่านทางการ์ดข้อมูล การจัดฟอร์แมตสำหรับบรรทัดข้อมูลต่าง ๆ ต้องปฏิบัติตามฟอร์แมตตามที่แสดงในตารางที่ ง.1 - ง.17 การเรียงลำดับของบรรทัดข้อมูล ต้องเรียงให้เป็นไปตามรูปที่ ง.1 สิ่งที่ต้องระวังคือการจัดฟอร์แมตข้อมูลของระบบเอกไซเตชัน เพราะระบบเอกไซเตชันที่โปรแกรม LFO สามารถนำไปใช้ประกอบการวิเคราะห์ได้ มีอยู่ 4 แบบ กล่าวคือระบบเอกไซเตชันชนิดที่ 1,2,3 และ 6 ในกรณีที่ไม่ต้องการใช้ระบบเอกไซเตชันของเครื่องกำเนิดไฟฟ้าใดมาประกอบการวิเคราะห์ ผู้ใช้ต้องลบบรรทัดข้อมูลของระบบเอกไซเตชันที่สัมนัยกับเครื่องกำเนิดไฟฟ้าดังกล่าวทิ้งไป ตัวอย่างเช่นถ้าต้องการวิเคราห์ระบบไฟฟ้ากำลังซึ่งประกอบด้วยเครื่องกำเนิดไฟฟ้า 5 เครื่อง และ แต่ละเครื่องต่างก็มีระบบเอกไซเตชัน บรรทัดข้อมูลของระบบเอกไซเตชันต้องมี 10 บรรทัด กล่าวคือเป็นบรรทัดข้อมูลแบบที่ 1 จำนวน 5 บรรทัด (ดูตารางที่ง.9 และ ง.10) และเป็นบรรทัดข้อมูลแบบที่ 2 อีก 5 บรรทัด (ดูตารางที่ ง.11, ง.12 และ ง.13) ถ้าไม่ต้องการนำเอาระบบเอกไซเตชันของเครื่องกำเนิดไฟฟ้าเครื่องใดมาประกอบการวิเคราะห์ ผู้ใช้ต้องลบบรรทัดข้อมูลที่สมนัยกับระบบเอกไซเตชันดังกล่าวทิ้งไป การจัดข้อมูลของอุปกรณ์สเตบิไลเซอร์ และ ข้อมูลของระบบควบคุมความเร็ว ก็ใช้หลักการการเดียวกันกับการจัดข้อมูลของระบบเอกไซเตชัน กล่าวคือเครื่อง

กำเนิดไฟฟ้าเครื่องใดไม่มีอุปกรณ์ดังกล่าว ก็ต้องลบบรรทัดข้อมูลของอุปกรณ์ ที่สมนัยกับเครื่องกำเนิดไฟฟ้าเครื่องนั้นทิ้งไป

ง.4 การแก้ไขข้อมูล

เมื่อโปรแกรมรับข้อมูลครบแล้วจะปรากฏข้อความ ตามที่ได้แสดงในกรอบข้างล่างขึ้นที่จอภาพ

```
MODIFY ANY DATA ? (Y/N) >
```

ถ้าผู้ใช้ต้องการแก้ไขข้อมูลบางส่วนให้พิมพ์ Y ถ้าไม่ต้องการแก้ไข ให้กด Return ในกรณีที่ผู้ใช้พิมพ์ Y จะปรากฏเมนู (menu) สำหรับการแก้ไขข้อมูลขึ้นที่จอภาพ โดยเริ่มจากเมนูที่แสดงในกรอบข้างล่าง

```
      CORRECTION MENU NO. 1
( 0) QUIT
( 1) TOTAL NUMBER OF BUSES
( 2) TOTAL NUMBER OF LINES
( 3) TOTAL NUMBER OF TRANSFORMERS
( 4) TOTAL NUMBER OF GENERATORS
( 5) FORCED GENERATOR
( 6) FREQUENCY OF POWER SYSTEM
( 7) BASE MVA OF SYSTEM
( 8) PER UNIT SPEED DEVIATION OF FORCED GENERATOR
( 9) MAXIMUM ITERATIONS FOR EIGENVALUE SEARCH
(10) TOLERANCE FOR EIGENVALUE SEARCH
(11) INITIAL INCREMENTAL FREQUENCY IN HERTZ
(12) LOWEST FREQUENCY OF FREQUENCY BAND IN HERTZ
```

```
      (FOR SWEEP FREQUENCY MODE)
(13) FREQUENCY INCREMENT IN HERTZ
      (FOR SWEEP FREQUENCY MODE)
(14) FREQUENCY INCREMENT IN HERTZ
      (FOR SWEEP FREQUENCY MODE)
      ENTER YOUR SELECTED NUMBER >
```

เมื่อผู้ใช้ต้องการแก้ไขข้อมูลใดในเมนูดังกล่าวนี้ ให้พิมพ์ระหัสของข้อมูลในเมนูลงไป จากนั้นข้อความที่เกี่ยวข้องกับข้อมูลดังกล่าวจะปรากฏขึ้นที่จอภาพเหมือนกับในตอนป้อนข้อมูลผ่านแป้นพิมพ์ ตัวอย่างเช่นถ้าผู้ใช้ต้องการแก้ข้อมูลหมายเลข 1 ในเมนูข้างต้น ผู้ใช้ต้องพิมพ์เลขหนึ่งต่อจากข้อความ ENTER YOUR SELECTED NUMBER > ที่ปรากฏขึ้นที่จอภาพ แล้วกด Return จากนั้นข้อความ TOTAL NUMBER OF BUSES = จะปรากฏขึ้นที่จอภาพ ผู้ใช้ต้องพิมพ์ค่าที่ถูกต้องของข้อมูลลงไปแล้วกด Return เมื่อกด Return แล้ว จะปรากฏเมนูเดิมขึ้นมาที่จอภาพใหม่เพื่อให้ผู้ใช้เลือกว่าจะแก้ไขข้อมูลใดอีก ถ้าผู้ใช้ไม่ต้องการแก้ไขข้อมูลในเมนูดังกล่าวนี้ ให้พิมพ์ 0 แล้วกด Return (หรือกด Return เลยก็ได้) จะปรากฏเมนูสำหรับแก้ไขข้อมูลเกี่ยวกับสายส่งซึ่งมีลักษณะตามที่ได้แสดงในกรอบข้างล่าง

```
      LINE DATA CORRECTION MENU
( 0) QUIT
( 1) STARTING BUS
( 2) ENDING BUS
( 3) LINE RESISTANCE
( 4) LINE REACTANCE
( 5) CHARGING MVA
      ENTER YOUR SELECTED NUMBER >
```

ถ้าผู้ใช้ต้องการแก้ไขข้อมูลเกี่ยวกับสายส่งตรงกับหมายเลขใดในเมนูดังกล่าวนี้ให้พิมพ์หมายเลขดังกล่าวต่อจากข้อความ ENTER YOUR SELECTED NUMBER > เช่นถ้าต้องการ

แก้ไขค่าของความต้านทานของสายส่ง ต้องเลือกหมายเลข 3 เมื่อพิมพ์ 3 ต่อจากข้อความ ENTER YOUR SELECTED NUMBER > แล้วกด Return จะปรากฏข้อความในกรอบข้างล่างขึ้นที่จอภาพ

```
PLEASE SPECIFY THE INDEX OF THAT LINE >
```

ผู้ใช้ต้องพิมพ์หมายเลขของสายส่งที่ต้องการแก้ไขค่าความต้านทานลงไป เช่นถ้าต้องการแก้ไขค่าความต้านทานของสายส่งหมายเลข 2 ก็พิมพ์เลข 2 ลงไป เมื่อกด Return แล้วจะปรากฏข้อความในกรอบข้างล่าง ขึ้นที่จอภาพ

```
RESISTANCE IN PU. OF LINE NO. 2  =
```

เมื่อปรากฏข้อความในกรอบข้างบนที่จอภาพแล้วผู้ใช้ต้องพิมพ์ค่าที่ถูกต้องของความต้านทานของสายส่งหมายเลข 2 ลงไป เมื่อกด Return แล้ว จะปรากฏเมนูเดิมขึ้นมาที่จอภาพใหม่เพื่อให้ผู้ใช้เลือกว่าจะแก้ไขข้อมูลใดอีก ถ้าผู้ใช้ไม่ต้องการแก้ไขข้อมูลในเมนูดังกล่าวนี้ให้กด Return จะปรากฏเมนูสำหรับแก้ไขข้อมูลเกี่ยวกับหม้อแปลงซึ่งมีลักษณะตามที่ได้แสดงในกรอบข้างล่าง

```
      TRANSFORMER DATA CORRECTION MENU
( 0) QUIT
( 1) PRIMARY SIDE BUS
( 2) SECONDARY SIDE BUS
( 3) TRANSFORMER RESISTANCE
( 4) TRANSFORMER REACTANCE
( 5) TAP SETTING
(6 ) PHASE SHIFTING
      ENTER YOUR SELECTED NUMBER >
```

การแก้ไขข้อมูลเกี่ยวกับหม้อแปลงทำได้เช่นเดียวกับการแก้ไขข้อมูลเกี่ยวกับสายส่ง ซึ่งได้อธิบายไปแล้ว เมนูที่จะปรากฏที่จอภาพต่อจากเมนูแก้ไขข้อมูลเกี่ยวกับหม้อแปลงคือเมนูแก้ไขข้อมูลของบัสซึ่งมีลักษณะตามที่ได้แสดงในกรอบข้างล่าง

```
      BUS DATA CORRECTION MENU
( 0) QUIT
( 1) REAL POWER GENERATION
( 2) REACTIVE POWER GENERATION
( 3) REAL POWER LOAD
( 4) REACTIVE POWER LOAD
( 5) MANITUDE OF VOLTAGE IN PU.
( 6) PHASE ANGLE OF VOLTAGE IN DEGRE
( 7) FIXED CAPACITANCE IN PU.
( 8) BUS TYPE CODE
      ENTER YOUR SELECTED NUMBER >
```

การแก้ไขข้อมูลเกี่ยวกับบัส มีวิธีการเช่นเดียวกับการแก้ไขข้อมูลเกี่ยวกับกับสายส่ง เมนูที่จะปรากฏที่จอภาพต่อจากเมนูแก้ไขข้อมูลเกี่ยวกับบัสคือเมนูแก้ไขข้อมูลเกี่ยวกับเครื่องกำเนิดไฟฟ้า ซึ่งมีลักษณะตามที่แสดงในกรอบข้างล่าง

```
    GENERATOR DATA CORRECTION MENU
( 0) QUIT
( 1) D - AXIS SYNCHRONOUS REACTANCE
( 2) D - AXIS TRANSIENT REACTANCE
( 3) D - AXIS SUBTRANSIENT REACTANCE
( 4) Q - AXIS SYNCHRONOUS REACTANCE
( 5) Q - AXIS TRANSIENT REACTANCE
( 6) Q - AXIS SUBTRANSIENT REACTANCE
```

```
( 7) LEAKAGE REACTANCE
( 8) ARMATURE RESISTANCE
( 9) POINTER REACTANCE
(10) SATURATION FUNCTION AT 1.0 PU. VOLTAGE
(11) SATURATION FUNCTION AT 1.2 PU. VOLTAGE
(12) INERTIA CONSTANCE
(13) MVA RATING
(14) DAMPING COEFFICIENT
(15) D - AXIS OPENNED CIRCUIT TRANSIENT TIME CONSTANCE
(16) D - AXIS OPENNED CIRCUIT SUBTRANSIENT TIME CONSTANCE
(17) Q - AXIS OPENNED CIRCUIT TRANSIENT TIME CONSTANCE
(18) Q - AXIS OPENNED CIRCUIT SUBTRANSIENT TIME CONSTANCE
(19) TERMINAL BUS OF GENERATOR
(20) EXCITER TYPE CODE
(21) POWER SYSTEM STABILIZER CODE
(22) SPEED GOVERNING SYSTEM CODE
     ENTER YOUR SELECTED NUMBER >
```

การแก้ไขข้อมูลเกี่ยวกับเครื่องกำเนิดไฟฟ้า มีวิธีการเช่นเดียวกับการแก้ไขข้อมูลเกี่ยวกับสายส่ง เมนูที่จะปรากฏที่จอภาพต่อจากเมนูแก้ไขข้อมูลเกี่ยวกับเครื่องกำเนิดไฟฟ้าคือเมนูแก้ไขข้อมูลเกี่ยวกับระบบเอกไซเตชันซึ่งมีลักษณะตามที่แสดงในกรอบข้างล่าง

```
  EXCITER DATA CORRECTION MENU
( 0) QUIT
( 1) TA
( 2) TB
( 3) TC
( 4) TE
```

```
( 5) TF
( 6) KA
( 7) KE
( 8) KF
( 9) SATURATION FUNCTION AT 1.0 PU. VOLTAGE
(10) SATURATION FUNCTION AT 1.2 PU. VOLTAGE
(11) VRMAX
(12) VRMIN
(13) KP
(14) KI
(15) VBMAX
(16) TRH
(17) TR
(18) COMPENSATING RESISTANCE
(19) COMPENSATING REACTANCE
(20) UPPER LIMIT OF EFD (EFDMAX)
(21) LOWER LIMIT OF EFD (EFDMIN)
     ENTER YOUR SELECTED NUMBER >
```

การแก้ไขข้อมูลเกี่ยวกับระบบเอกไซเตชัน มีวิธีการเช่นเดียวกับการแก้ไข้ข้อมูลเกี่ยวกับสายส่ง เมนูที่จะปรากฏที่จอภาพต่อจากเมนูแก้ไขข้อมูลเกี่ยวกับระบบเอกไซเตชันคือเมนูแก้ไขข้อมูลเกี่ยวกับอุปกรณ์สเตบิไลเซอร์ซึ่งมีลักษณะตามที่แสดงในกรอบข้างล่าง

```
     POWER SYSTEM STABILIZER DATA CORRECTION MENU
( 0) QUIT
( 1) K1
( 2) T1
( 3) T3
```

```
( 4) T4
( 5) T5
( 6) T6
( 7) T7
( 8) T8
( 9) L1
(10) L2
     ENTER YOUR SELECTED NUMBER >
```

การแก้ไขข้อมูลเกี่ยวกับอุปกรณ์สเตบิไลเซอร์ มีวิธีการเช่นเดียวกับการแก้ไขข้อมูลเกี่ยวกับสายส่ง เมนูที่จะปรากฏที่จอภาพต่อจากเมนูแก้ไขข้อมูลเกี่ยวกับอุปกรณ์สเตบิไลเซอร์ คือเมนูแก้ไขข้อมูลเกี่ยวกับระบบควบคุมความเร็วซึ่งมีลักษณะตามที่แสดงในกรอบข้างล่าง

```
     SPEED-GOVERNING SYSTEM DATA CORRECTION MENU
( 0) QUIT
( 1) K1
( 2) K2
( 3) K3
( 4) K4
( 5) T1
( 6) T2
( 7) T3
( 8) T4
( 9) T5
(10) T6
     ENTER YOUR SELECTED NUMBER >
```

การแก้ไขข้อมูลเกี่ยวระบบควบคุมความเร็ว มีวิธีการเช่นเดียวกับการแก้ไขข้อมูลเกี่ยวกับสายส่ง

ง.5 ไฟล์ข้อมูลสำรอง

ในการทำงานแต่ละครั้งของโปรแกรม LFO มันได้สร้างไฟล์ชื่อ LFO.SAV ขึ้นมา ไฟล์ดังกล่าวนี้ ได้รับการสร้างให้อยู่ในดิสค์ (disk) แผ่นเดียวกับโปรแกรม LFO ไฟล์ LFO.SAV นี้เป็นไฟล์บันทึกข้อมูลในการทำงานแต่ละครั้งโปรแกรม LFO ข้อมูลในไฟล์ LFO.SAV มีการจัดฟอร์แมตของข้อมูลถูกต้องตามหลักการจัดฟอร์แมตสำหรับโปรแกรม LFO ทุกประการ ถ้าในการทำงานครั้งแรกของโปรแกรม LFO ผู้ใช้ป้อนข้อมูลผ่านทางแป้นพิมพ์ ในการทำงานครั้งต่อไป ถ้าผู้ใช้ต้องการนำเอาข้อมูลดังกล่าวมาใช้อีก ผู้ใช้เพียงแต่ถ่าย (copy) ข้อมูลจากไฟล์ LFO.SAV ลงในไฟล์ LFO.DAT โปรแกรมฯ จะนำเอาข้อมูลจากไฟล์ LFO.DAT นี้ไปใช้ในการทำงานครั้งต่อ ๆ ไป ถ้าหากผู้ใช้เลือกโมดของการป้อนข้อมูลโดยให้โปรแกรมฯ อ่านข้อมูลจากไฟล์ข้อมูล

ง.6 ไฟล์บันทึกคำตอบของโปรแกรม LFO

ในการทำงานแต่ละครั้งของโปรแกรม LFO มันได้บันทึกข้อมูลที่ใช้ในการทำงานและคำตอบที่มันหาได้ ลงในไฟล์ LFO.OP

คอลัมน์	ฟอร์แมต	ข้อมูล
1 - 5	I5	จำนวนบัสของระบบไฟฟ้ากำลัง
6 - 10	I5	จำนวนสายส่งของระบบไฟฟ้ากำลัง
11 - 15	I5	จำนวนหม้อแปลงของระบบไฟฟ้ากำลัง
16 - 20	I5	จำนวนเครื่องกำเนิดไฟฟ้าของระบบไฟฟ้ากำลัง
21 - 25	I5	จำนวนวงรอบการค้นหาค่าไอเก็นที่ยอมได้
26 - 30	I5	หมายเลขของเครื่องกำเนิดไฟฟ้าซึ่งถูกบังคับ
31 - 35	I5	จำนวนครั้งของการคำนวณ ตามวิธีการ สวีพฟรีเควันซี (sweep frequency)

ตารางที่ ง.1 ฟอร์แมตของบรรทัดข้อมูลแบบที่ 1

คอลัมน์	ฟอร์แมต	ข้อมูล
1 - 10	F10.4	ความถี่ของระบบไฟฟ้ากำลัง
11 - 20	F10.4	ค่าเบสของ MVA
21 - 30	F10.4	ค่าความคลาดเคลื่อนที่ยอมให้ได้ ในการค้นหาค่า λ กล่าวคือ $\lvert\lambda_{new} - \lambda_{old}\rvert \leq \epsilon$
31 - 40	F10.4	ค่าความเร็วเชิงมุมเบี่ยงเบนของเครื่องกำเนิดไฟฟ้าซึ่งถูกบังคับ (เปอร์ยูนิต)
41 - 50	F10.4	ความถี่เริ่มต้น f_o ในการค้นหาค่าไอเก็น ในหน่วยเฮิร์ตซ์ การค้นหาจะเริ่มจากจุด $\lambda_o = \sigma_o + j\Omega_o = 0 + j2\Pi f_o$
51 - 60	F10.4	ค่าความถี่เริ่มต้นในหน่วยเฮิร์ตซ์ ตามวิธีการสวีพฟรีเควันซี
61 - 70	F10.4	ค่าส่วนเพิ่มของความถี่ (frequency increment) ตามวิธีการสวีพฟรีเควันซี

ตารางที่ ง.2 ฟอร์แมตของบรรทัดข้อมูลแบบที่ 2

คอลัมน์	ฟอร์แมต	ข้อมูล
1 - 5	I5	บัสที่จุดต้นทางของสายส่ง
6 - 10	I5	บัสที่จุดปลายทางของสายส่ง
11 - 20	F10.4	ความต้านทานของสายส่ง (เปอร์ยูนิต)
21 - 30	F10.4	รีแอกแตนซ์ของสายส่ง (เปอร์ยูนิต)
31 - 40	F10.4	Line Charging ของสายส่งที่ระดับแรงดัน 1.0 เปอร์ยูนิต ในหน่วย MVAR

ตารางที่ ง.3 ฟอร์แมตของบรรทัดข้อมูลของสายส่ง

คอลัมน์	ฟอร์แมต	ข้อมูล
1 - 5	I5	บัสปฐมภูมิ (primary bus) ของหม้อแปลง
6 - 10	I5	บัสทุติยภูมิ (secondary bus) ของหม้อแปลง
11 - 20	F10.4	ความต้านทานของหม้อแปลง (เปอร์ยูนิต)
21 - 30	F10.4	รีแอกแตนซ์ของหม้อแปลง (เปอร์ยูนิต)
31 - 40	F10.4	อัตราการแท็ป (tap) ของหม้อแปลง ยกตัวอย่างเช่น หม้อแปลงเป็นชนิดแท็ป 5.0 % ต้องป้อนข้อมูล 1.05
41 - 50	F10.4	การเลื่อนเฟส (phase-shift) ของหม้อแปลงเป็นองศา ในกรณีที่หม้อแปลงเป็นหม้อแปลงเลื่อนเฟส

ตารางที่ ง.4 ฟอร์แมตของบรรทัดข้อมูลของหม้อแปลง

คอลัมน์	ฟอร์แมต	ข้อมูล
1 - 10	F10.4	กำลังไฟฟ้าจริงที่จ่ายเข้าที่บัส ในหน่วย MW
11 - 20	F10.4	กำลังไฟฟ้าจริงของโหลด ในหน่วย MW
21 - 30	F10.4	กำลังไฟฟ้ารีแอกตีฟที่จ่ายเข้าที่บัส ในหน่วย MVAR
31 - 40	F10.4	กำลังไฟฟ้ารีแอกตีฟของโหลด ในหน่วย MVAR
41 - 50	F10.4	ขนาดของแรงดันที่บัส (เปอร์ยูนิต)
51 - 60	F10.4	มุมของแรงดันที่บัส ในหน่วยดีกรี
61 - 70	F10.4	ค่าความจุไฟฟ้า (capacitance) ของคาพาซิเตอร์ (capacitor) ซึ่งต่ออยู่ที่บัส (เปอร์ยูนิต)
71 - 75	I5	ระหัสสำหรับบอกชนิดของบัส ป้อนค่า "1" สำหรับบัสของเครื่องกำเนิดไฟฟ้า (generator bus) ป้อนค่า "2" สำหรับบัสของโหลด (load bus) ป้อนค่า "3" สำหรับ อินฟินิตบัส (infinite bus)

ตารางที่ ง.5 ฟอร์แมตของบรรทัดข้อมูลบัส

คอลัมน์	ฟอร์แมต	ข้อมูล
1 - 10	F10.4	R_A (เปอร์ยูนิต)
11 - 20	F10.4	X_1 (เปอร์ยูนิต)
21 - 30	F10.4	X_D (เปอร์ยูนิต)
31 - 40	F10.4	X'_D (เปอร์ยูนิต)
41 - 50	F10.4	X''_D (เปอร์ยูนิต)
51 - 60	F10.4	X_Q (เปอร์ยูนิต)
61 - 70	F10.4	X'_Q (เปอร์ยูนิต)
71 - 80	F10.4	X''_Q (เปอร์ยูนิต)

ตารางที่ ง.6 ฟอร์แมตของบรรทัดข้อมูลแบบที่ 1 ของเครื่องกำเนิดไฟฟ้า

คอลัมน์	ฟอร์แมต	ข้อมูล
1 - 10	F10.4	T'_{DO}
11 - 20	F10.4	T''_{DO}
21 - 30	F10.4	T'_{QO}
31 - 40	F10.4	T''_{QO}
41 - 50	F10.4	H (เปอร์ยูนิต)
51 - 60	F10.4	พิกัด MVA
61 - 70	F10.4	D (ค่าคงตัวของการหน่วง)

ตารางที่ ง.7 ฟอร์แมตของบรรทัดข้อมูลแบบที่ 2 ของเครื่องกำเนิดไฟฟ้า

คอลัมน์	ฟอร์แมต	ข้อมูล
11 - 20	F10.4	ค่าของฟังก์ชันการอิ่มตัวที่ระดับแรงดัน 1.0 เปอร์ยูนิต (ดูรูปที่ 4.8)
21 - 30	F10.4	ค่าของฟังก์ชันการอิ่มตัวที่ระดับแรงดัน 1.2 เปอร์ยูนิต (ดูรูปที่ 4.8)
31 - 35	I5	ระหัสบอกชนิดของระบบเอกไซเตชัน (ดูตารางที่ 5.1)
36 - 40	I5	ระหัสสำหรับบอกว่ามีอุปกรณ์สเตบิไลเซอร์หรือไม่ ("1" หมายถึงมี "0" หมายถึงไม่มี)
41 - 45	I5	ระหัสสำหรับบอกว่ามีระบบควบคุมความเร็วหรือไม่ ("1" หมายถึงมี "0" หมายถึงไม่มี)
46 - 50	I5	บัสที่ขั้วของเครื่องกำเนิดไฟฟ้า

ตารางที่ ง.8 ฟอร์แมตของบรรทัดข้อมูลแบบที่ 3 ของเครื่องกำเนิดไฟฟ้า

คอลัมน์	ฟอร์แมต	ข้อมูล
1 - 10	F10.4	K_A
11 - 20	F10.4	T_A
21 - 30	F10.4	V_{RMAX}
31 - 40	F10.4	V_{RMIN}
41 - 50	F10.4	R_C ($Z_C = R_C + jX_C$ ดูรูปที่ 5.2)
51 - 60	F10.4	X_C ($Z_C = R_C + jX_C$ ดูรูปที่ 5.2)
61 - 70	F10.4	T_B
71 - 80	F10.4	T_C

ตารางที่ ง.9 ฟอร์แมตของบรรทัดข้อมูลแบบที่ 1
ของระบบเอกไซเตชันชนิดที่ 1, 2, และ 6

คอลัมน์	ฟอร์แมต	ข้อมูล
1 - 10	F10.4	K_A
11 - 20	F10.4	T_A
21 - 30	F10.4	V_{RMAX}
31 - 40	F10.4	V_{RMIN}
41 - 50	F10.4	R_C ($Z_C = R_C + jX_C$ ดูรูปที่ 5.2)
51 - 60	F10.4	X_C ($Z_C = R_C + jX_C$ ดูรูปที่ 5.2)

ตารางที่ ง.10 ฟอร์แมตของบรรทัดข้อมูลแบบที่ 1 ของระบบเอกไซเตชันชนิดที่ 3

คอลัมน์	ฟอร์แมต	ข้อมูล
1 - 10	F10.4	K_E
11 - 20	F10.4	T_E
21 - 30	F10.4	S_1
31 - 40	F10.4	S_2
41 - 50	F10.4	K_F
51 - 60	F10.4	T_F
61 - 70	F10.4	T_R

ตารางที่ ง.11 ฟอร์แมตของบรรทัดข้อมูลแบบที่ 2 ของระบบเอกไซเตชันชนิดที่ 1 และ 2

คอลัมน์	ฟอร์แมต	ข้อมูล
1 - 10	F10.4	K_E
11 - 20	F10.4	T_E
21 - 30	F10.4	K_P
31 - 40	F10.4	K_I
41 - 50	F10.4	V_{BMAX}
51 - 60	F10.4	K_F
61 - 70	F10.4	T_F
71 - 80	F10.4	T_R

ตารางที่ ง.12 ฟอร์แมตของบรรทัดข้อมูลแบบที่ 2 ของระบบเอกไซเตชันชนิดที่ 3

คอลัมน์	ฟอร์แมต	ข้อมูล
41 - 50	F10.4	K_F
51 - 60	F10.4	T_F
61 - 70	F10.4	T_R

ตารางที่ ง.13 ฟอร์แมตของบรรทัดข้อมูลแบบที่ 2 ของระบบเอกไซเตชันชนิดที่ 6

คอลัมน์	ฟอร์แมต	ข้อมูล
1 - 10	F10.4	K_1
11 - 20	F10.4	T_1
21 - 30	F10.4	T_3
31 - 40	F10.4	T_4 (ดูรูปที่ 5.9)
41 - 50	F10.4	T_5
51 - 60	F10.4	T_6
61 - 70	F10.4	T_7
71 - 80	F10.4	T_8

ตารางที่ ง.14 ฟอร์แมตของบรรทัดข้อมูลแบบที่ 1 ของอุปกรณ์สเตบิไลเซอร์

คอลัมน์	ฟอร์แมต	ข้อมูล
1 - 10	F10.4	L_1
11 - 20	F10.4	L_2 (ดูรูปที่ 5.9)

ตารางที่ ง.15 ฟอร์แมตของบรรทัดข้อมูลแบบที่ 2 ของอุปกรณ์สเตบิไลเซอร์

คอลัมน์	ฟอร์แมต	ข้อมูล
1 - 10	F10.4	K_1
11 - 20	F10.4	T_1
21 - 30	F10.4	T_2
31 - 40	F10.4	T_3
		(ดูรูปที่ 5.10)

ตารางที่ ง.16 ฟอร์แมตของบรรทัดข้อมูลแบบที่ 1 ของระบบควบคุมความเร็ว

คอลัมน์	ฟอร์แมต	ข้อมูล
1 - 10	F10.4	K_2
11 - 20	F10.4	T_4
21 - 30	F10.4	T_5
31 - 40	F10.4	K_3 (ดูรูปที่ 5.10)
41 - 50	F10.4	K_4
51 - 60	F10.4	T_6

ตารางที่ ง.17 ฟอร์แมตของบรรทัดข้อมูลแบบที่ 1 ของระบบควบคุมความเร็ว

บรรทัดข้อมูลทั่วไปลำดับที่ 1 (บรรทัดที่ 1)
บรรทัดข้อมูลทั่วไปลำดับที่ 2 (บรรทัดที่ 2)
บรรทัดข้อมูลสายส่ง (เท่ากับจำนวนสายส่ง)
บรรทัดข้อมูลหม้อแปลง (เท่ากับจำนวนหม้อแปลง)
บรรทัดข้อมูลบัส (เท่ากับจำนวนบัส)
บรรทัดข้อมูลเครื่องกำเนิดไฟฟ้าแบบที่ 1 (เท่ากับจำนวนเครื่องกำเนิดไฟฟ้า)
บรรทัดข้อมูลเครื่องกำเนิดไฟฟ้าแบบที่ 2 (เท่ากับจำนวนเครื่องกำเนิดไฟฟ้า)
บรรทัดข้อมูลเครื่องกำเนิดไฟฟ้าแบบที่ 3 (เท่ากับจำนวนเครื่องกำเนิดไฟฟ้า)
บรรทัดข้อมูลระบบเอกไซเตชันแบบที่ 1 (เฉพาะเครื่องกำเนิดไฟฟ้าที่มีระบบเอกไซเตชันเท่านั้น)
บรรทัดข้อมูลระบบเอกไซเตชันแบบที่ 2 (เฉพาะเครื่องกำเนิดไฟฟ้าที่มีระบบเอกไซเตชันเท่านั้น)
บรรทัดข้อมูลอุปกรณ์สเตบิไลเซอร์แบบที่ 1 (เฉพาะเครื่องกำเนิดไฟฟ้าที่มี อุปกรณ์สเตบิไลเซอร์ เท่านั้น)
บรรทัดข้อมูลอุปกรณ์สเตบิไลเซอร์แบบที่ 2 (เฉพาะเครื่องกำเนิดไฟฟ้าที่มี อุปกรณ์สเตบิไลเซอร์ เท่านั้น)
บรรทัดข้อมูลระบบควบคุมความเร็วแบบที่ 1 (เฉพาะเครื่องกำเนิดไฟฟ้าที่มีระบบควบคุมความเร็วเท่านั้น)
บรรทัดข้อมูลระบบควบคุมความเร็วแบบที่ 2 (เฉพาะเครื่องกำเนิดไฟฟ้าที่มีระบบควบคุมความเร็วเท่านั้น)

รูปที่ ง.1 การจัดลำดับบรรทัดข้อมูลในไฟล์ข้อมูล

ภาคผนวก จ

โปรแกรมวิเคราะห์ออสซิลเลชันความถี่ต่ำที่เกิดขึ้นเองในระบบไฟฟ้ากำลัง

```
C     MAIM PROGRAM
      COMMON /CX1/ANG(10),BW(10),BX(10),CUI(10),CUR(10),DAMP(10),
     *DEL(10),DELV1(10),DELV2(10),EFDMAX(10),EFDMIN(10),EGD(10),
     *EGQ(10),GA(10),GC(10),GI(10),GP(10),GV(10),GW(10),GX(10),
     *G1(10),G2(10),G3(10),G4(10),HBASE(10),HC(10),P(15),PL(15),
     *PM(15),PHSF(10),Q(15),QL(15),RA(10),RL(10),RT(10),RV(10),
     *S1(10),S2(10),SAT1(10),SAT2(10),TA(10),TA2(10),TA3(10),
     *TAP(10),TDOP(10),TDOPP(10),TE(10),TF(10),TQOP(10),TQOPP(10),
     *TR(10),T1(10),T2(10),T3(10),T4(10),T5(10),T6(10),VB(15),
     *VBMAX(10),VHMAX(10),VHMIN(10),VRMAX(10),VRMIN(10),VL1(10),
     *VL2(10),XD(10),XDP(10),XDPP(10),XKE(10),XKEP(10),XKF(10),
     *XKFP(10),XL(10),XP(10),XQ(10),XQP(10),XQPP(10),XS(10),XT(10),
     *XV(10),XG1(10),XT1(10),XT3(10),XT4(10),XT5(10),XT6(10),XT7(10),
     *XT8(10),YSHUNT(10)
      INTEGER GB(5),TYPEX(5),SB(20),EB(20),ST(10),ET(10),IMAG(20),
     *TYPE(20),LIST(40),NEXT(800),FAR(800),NCONN(40),MODEL(8),
     *IPSS(10),IGOV(10)
      COMPLEX Z(10,10),YCHARG(20),SPD(30),RH(30),SPC(30,4),TC(20),
     *EE(30),E(15),ASELF(40),AMUT(800)
      COMMON /CX2/GB,TYPEX,SB,EB,ST,ET,IMAG,TYPE,LIST,NEXT,FAR,NCONN,
     *MODEL,IPSS,IGOV
      COMMON /CX3/Z,YCHARG,SPD,RH,SPC,TC,EE,E,ASELF,AMUT
      COMMON /CX4/REXC,XEXC,RGOV,XGOV,RPSS,XPSS,FREQCY,C1DS,C1QS,C2DS,
     *C2QS,CV1,CV2,CV3,CV4,EFDO,SIGMA,OMEGA,TOL,PI,WD,FMIN,FINC,FO,
     *SBASE,Y11,Y12,Y21,Y22,C1D,C2D,C3D,C4D,C5D,C6D,C7D,C1Q,C2Q,C3Q,
     *C4Q,C5Q,C6Q,C7Q,TSDO,TFDO,TSQO,TFQO,QAXISX,DAXISX
      COMMON /CX5/NB,NL,NT,NG,MAX,MAXF,IDRIVE,IC,ISEL
      OPEN(5,FILE='LFO.DAT',STATUS='OLD')
      OPEN(6,FILE='LFO.OP',STATUS='NEW')
      OPEN(7,FILE='DAT.SAVE',STATUS='NEW')
      CALL SELECT(IP)
      CALL FIRST(PI,P,PL,Q,QL,YSHUNT,VB,ANG,SBASE,E,
     *NB,NL,FO,SIGMA,OMEGA)
      CALL GMODEL(NG,SBASE,HBASE,RA,XD,XQ,XDP,XQP,XDPP,
```

```
     *XQPP,XL,TDOP,TQOP,TDOPP,TQOPP,HC,DAMP,MODEL)
      CALL INTIAL(NG,RA,XD,XQ,XDP,XQP,XDPP,XQPP,XL,
     *MODEL,P,Q,PM,GB,DEL,CUR,CUI,EGD,EGQ,SAT1,SAT2,XP,E)
      CALL CCC(NB,NG,IMAG,GB)
      CALL YCH(NB,PL,QL,YCHARG,YSHUNT,SB,EB,ST,ET,NL,NT,E,
     *RT,XT,TAP,PHSF,RL,XS,TYPE)
      DO 1 K        =        1,NG
      CALL CONSTA(K,RA,XD,XQ,XDP,XQP,XDPP,XQPP,XL,TDOP,TQOP,
     *TDOPP,TQOPP,MODEL)
      CALL ACONST(K,RA,XD,XQ,XDP,XQP,XDPP,XQPP,XL,
     *MODEL,P,Q,PM,GB,DEL,CUR,CUI,EGD,EGQ,SAT1,SAT2,XP,E,
     *TYPEX,RV,XV,Z)
      CALL EXCITER(K,TYPEX,GB,GA,TA,VRMIN,VRMAX,RV,XV,TA2,TA3,
     *XKE,TE,S1,S2,EFDMAX,EFDMIN,XKF,TF,TR,XKEP,XKFP,DELV1,DELV2,Q,GP,
     *GI,GC,GV,CUR,CUI,VBMAX)
      CALL GGG(K,GW,GX,BW,BX)
    1 CONTINUE
      CALL SEARC(IP)
      CLOSE(5)
      CLOSE(6)
      CLOSE(7)
      STOP
      END
C     SUBROUTINE SEARC
      SUBROUTINE SEARC(IP)
      COMMON /CX1/ANG(10),BW(10),BX(10),CUI(10),CUR(10),DAMP(10),
     *DEL(10),DELV1(10),DELV2(10),EFDMAX(10),EFDMIN(10),EGD(10),
     *EGQ(10),GA(10),GC(10),GI(10),GP(10),GV(10),GW(10),GX(10),
     *G1(10),G2(10),G3(10),G4(10),HBASE(10),HC(10),P(15),PL(15),
     *PM(15),PHSF(10),Q(15),QL(15),RA(10),RL(10),RT(10),RV(10),
     *S1(10),S2(10),SAT1(10),SAT2(10),TA(10),TA2(10),TA3(10),
     *TAP(10),TDOP(10),TDOPP(10),TE(10),TF(10),TQOP(10),TQOPP(10),
     *TR(10),T1(10),T2(10),T3(10),T4(10),T5(10),T6(10),VB(15),
     *VBMAX(10),VHMAX(10),VHMIN(10),VRMAX(10),VRMIN(10),VL1(10),
     *VL2(10),XD(10),XDP(10),XDPP(10),XKE(10),XKEP(10),XKF(10),
     *XKFP(10),XL(10),XP(10),XQ(10),XQP(10),XQPP(10),XS(10),XT(10),
     *XV(10),XG1(10),XT1(10),XT3(10),XT4(10),XT5(10),XT6(10),XT7(10),
     *XT8(10),YSHUNT(10)
```

```
      INTEGER GB(5),TYPEX(5),SB(20),EB(20),ST(10),ET(10),IMAG(20),
     *TYPE(20),LIST(40),NEXT(800),FAR(800),NCONN(40),MODEL(8),
     *IPSS(10),IGOV(10)
      COMPLEX Z(10,10),YCHARG(20),SPD(30),RH(30),SPC(30,4),TC(20),
     *EE(30),E(15),ASELF(40),AMUT(800)
      COMMON /CX2/GB,TYPEX,SB,EB,ST,ET,IMAG,TYPE,LIST,NEXT,FAR,NCONN,
     *MODEL,IPSS,IGOV
      COMMON /CX3/Z,YCHARG,SPD,RH,SPC,TC,EE,E,ASELF,AMUT
      COMMON /CX4/REXC,XEXC,RGOV,XGOV,RPSS,XPSS,FREQCY,C1DS,C1QS,C2DS,
     *C2QS,CV1,CV2,CV3,CV4,EFDO,SIGMA,OMEGA,TOL,PI,WD,FMIN,FINC,FO,
     *SBASE,Y11,Y12,Y21,Y22,C1D,C2D,C3D,C4D,C5D,C6D,C7D,C1Q,C2Q,C3Q,
     *C4Q,C5Q,C6Q,C7Q,TSDO,TFDO,TSQO,TFQO,QAXISX,DAXISX
      COMMON /CX5/NB,NL,NT,NG,MAX,MAXF,IDRIVE,IC,ISEL
      IF(ISEL.EQ.1)GO TO 2
      WRITE(IP,1)
    1 FORMAT(1H ,3X,72('-'),/,4X,'I',12X,
     *'S W E E P  -  F R E Q U E N C Y',
     *'   O U T P U T',13X,'I',/,4X,72('-'),/,4X,'I FREQ. I PU EXTERNAL'
     *,' TORQUE I PEAK KINETIC I   ESTIMATED EIGENVALUE   I',/,4X,72
     *('-'),/,4X,'I HERTZ I  HORIZ   I  VERTI  I  ENERGY(MJ)  I  ALPHA '
     *,'I  BETA  I  HERTZ I',/,4X,72('-'))
      FNOW          =       FMIN
      SIGMA         =       0.0
      OMEGA         =       PI*FMIN*2.0
    2 IX            =       1
      NX            =       (NB + NG)*2
      IF(ISEL.EQ.1)WRITE(IP,3)
    3 FORMAT(1H ,3X,72('-'),/,4X,'I          C O M P L E X   P O L E',
     *'   S E A R C H   O U T P U T          I',
     */,4X,72('-'),/,4X,'I          ESTIMATED EIGENVALUE       I',
     *'    EXTERNAL TORQUE    I   ITER   I',/,4X,72('-'),/,4X,
     *'I   ALPHA   I    BETA   I   HERTZ   I   HORIZ   I   VERTI   I',
     *'    NO.   I',/,4X,72('-'))
    4 LAST          =       0
      DO 5 I        =       1,NX
      LIST(I)       =       0
      ASELF(I)      =       CMPLX(0.0,0.0)
    5 CONTINUE
```

```
      DO 6 K          =          1,NG
       CALL CONSTA(K,RA,XD,XQ,XDP,XQP,XDPP,XQPP,XL,TDOP,TQOP,
     *TDOPP,TQOPP,MODEL)
       CALL ACONST(K,RA,XD,XQ,XDP,XQP,XDPP,XQPP,XL,
     *MODEL,P,Q,PM,GB,DEL,CUR,CUI,EGD,EGQ,SAT1,SAT2,XP,E,
     *TYPEX,RV,XV,Z)
       CALL GNEXC(K,GA,TA3,TA,TA2,XKEP,XKFP,TE,TF,TYPEX)
       CALL GXGOV(K,IGOV,G1,G2,G3,G4,T1,T2,T3,T4,T5,T6)
       CALL GXPSS(K,IPSS,XG1,XT1,XT3,XT4,XT5,XT6,XT7,XT8,
     *TR,VL1,VL2)
       CALL ZCONST(K,Z,TR,DAMP,HC)
       KK              =          5
       IF(K.EQ.IDRIVE)KK=6
       CALL ELILIM1(Z,K,KK,10)
       CALL SPEED(Z,K,SPC)
       IF(K.EQ.IDRIVE)CALL RIGH(NB,NG,IDRIVE,Z,RH,WD)
       IF(K.EQ.IDRIVE)CALL TORQUE(TC,Z)
       CALL ARM2(K,NB,LIST,NEXT,FAR,ASELF,AMUT,GW,GX,BW,BX,
     *LAST,Z)
    6 CONTINUE
       NX              =          (NB+NG)*2
       CALL ARM1(NB,NL,NT,NG,SB,EB,ST,ET,RL,XS,RT,XT,TAP,PHSF,
     *YCHARG,ASELF,AMUT,GW,BW,GX,BX,LIST,NEXT,FAR,IMAG,GB,TYPE,LAST)
       CALL ELIMIN(RH,EE,ASELF,AMUT,LIST,NEXT,FAR,NX,LAST,
     *NCONN)
       NN              =          NG*2 + 1
       MM              =          NB*2
       CALL TT2(NB,TC,IDRIVE,EE,TXV,TXH,WD)
       CALL SPEED2(SPC,SPD,EE,NB,NG,IDRIVE)
       CALL CHECK(NG,SPD,SIGMA,OMEGA,WD,IDRIVE,HC,ALPHA,BETA,
     *TXV,TXH,WK)
       IF(ISEL.EQ.2)GO TO 14
       XX              =          BETA/(PI*2.0)
       WRITE(IP,7)ALPHA,BETA,XX,TXH,TXV,IX
    7 FORMAT(1H  ,3X,3('I  ',F7.4,2X),2('I  ',F7.5,2X),'I',
     *4X,I2,4X,'I')
       A               =          ABS(ALPHA-SIGMA)
       B               =          ABS(BETA-OMEGA)
```

```
      IF(A.LE.TOL.AND.B.LE.TOL)GO TO 8
      SIGMA          =       ALPHA
      OMEGA          =       BETA
      IX             =       IX + 1
      IF(IX.LE.MAX)GO TO 4
      GO TO 12
    8 CONTINUE
      WRITE(IP,9)GB(IDRIVE)
    9 FORMAT(1H ,3X,72('-'),/,4X,'I',19X,'DRIVEN MACHINE IS ON BUS NO.'
     *,2X,I2,19X,'I',/,4X,72('-'))
      WRITE(IP,99)
   99 FORMAT(1H ,3X,60('-'),/,4X,'I',18X,'FINAL GENERATOR SPEEDS',18X,
     *'I',/,4X,60('-'),/,4X,'I     BUS    I      HORIZONTAL      I',
     *'       VERTICAL       I',/,4X,60('-'))
      DO 11 I        =       1,NG
      WRITE(IP,10)GB(I),REAL(SPD(I)),AIMAG(SPD(I))
   10 FORMAT(1H ,3X,'I',5X,I2,5X,'I',2(5X,F10.6,7X,'I'))
   11 CONTINUE
      WRITE(IP,83)
   83 FORMAT(1H ,3X,60('-'))
      GO TO 16
   12 WRITE(*,13)
   13 FORMAT(1H ,3X,'EIGENVALUE SEARCH DIVERGS')
      GO TO 16
   14 CONTINUE
      IX             =       IX + 1
      IF(IX.GT.MAXF)GO TO 16
      HZ             =       BETA/(PI*2.0)
      WRITE(IP,15)FNOW,TXH,TXV,WK*SBASE,-ALPHA,BETA,HZ
   15 FORMAT(1H ,3X,'I ',F5.3,' I ',F7.4,'  I ',F6.4,'  I   ',F7.4,
     *4X,'I',3(1X,F6.4,1X,'I'))
      SIGMA          =       0.0
      FNOW           =       FNOW + FINC
      OMEGA          =       FNOW*PI*2.0
      GO TO 4
   16 CONTINUE
      IF(ISEL.EQ.2)WRITE(IP,17)WD,GB(IDRIVE)
   17 FORMAT(1H ,3X,72('-'),/,4X,'I',15X,'EXTERNAL TORQUE HOLD ',F5.3,
```

```
     *' PU AT BUS ',I2,16X,'I',/,4X,72('-'))
      RETURN
      END
C     SUBROUTINE SELECT
      SUBROUTINE SELECT(IP)
      COMMON /CX1/ANG(10),BW(10),BX(10),CUI(10),CUR(10),DAMP(10),
     *DEL(10),DELV1(10),DELV2(10),EFDMAX(10),EFDMIN(10),EGD(10),
     *EGQ(10),GA(10),GC(10),GI(10),GP(10),GV(10),GW(10),GX(10),
     *G1(10),G2(10),G3(10),G4(10),HBASE(10),HC(10),P(15),PL(15),
     *PM(15),PHSF(10),Q(15),QL(15),RA(10),RL(10),RT(10),RV(10),
     *S1(10),S2(10),SAT1(10),SAT2(10),TA(10),TA2(10),TA3(10),
     *TAP(10),TDOP(10),TDOPP(10),TE(10),TF(10),TQOP(10),TQOPP(10),
     *TR(10),T1(10),T2(10),T3(10),T4(10),T5(10),T6(10),VB(15),
     *VBMAX(10),VHMAX(10),VHMIN(10),VRMAX(10),VRMIN(10),VL1(10),
     *VL2(10),XD(10),XDP(10),XDPP(10),XKE(10),XKEP(10),XKF(10),
     *XKFP(10),XL(10),XP(10),XQ(10),XQP(10),XQPP(10),XS(10),XT(10),
     *XV(10),XG1(10),XT1(10),XT3(10),XT4(10),XT5(10),XT6(10),XT7(10),
     *XT8(10),YSHUNT(10)
      INTEGER GB(5),TYPEX(5),SB(20),EB(20),ST(10),ET(10),IMAG(20),
     *TYPE(20),LIST(40),NEXT(800),FAR(800),NCONN(40),MODEL(8),
     *IPSS(10),IGOV(10)
      COMPLEX Z(10,10),YCHARG(20),SPD(30),RH(30),SPC(30,4),TC(20),
     *EE(30),E(15),ASELF(40),AMUT(800)
      COMMON /CX2/GB,TYPEX,SB,EB,ST,ET,IMAG,TYPE,LIST,NEXT,FAR,NCONN,
     *MODEL,IPSS,IGOV
      COMMON /CX3/Z,YCHARG,SPD,RH,SPC,TC,EE,E,ASELF,AMUT
      COMMON /CX4/REXC,XEXC,RGOV,XGOV,RPSS,XPSS,FREQCY,C1DS,C1QS,C2DS,
     *C2QS,CV1,CV2,CV3,CV4,EFDO,SIGMA,OMEGA,TOL,PI,WD,FMIN,FINC,FO,
     *SBASE,Y11,Y12,Y21,Y22,C1D,C2D,C3D,C4D,C5D,C6D,C7D,C1Q,C2Q,C3Q,
     *C4Q,C5Q,C6Q,C7Q,TSDO,TFDO,TSQO,TFQO,QAXISX,DAXISX
      COMMON /CX5/NB,NL,NT,NG,MAX,MAXF,IDRIVE,IC,ISEL
      CHARACTER OK
      WRITE(*,50)
   50 FORMAT(1H ,3X,67('*'),/,4X,'*',2X,'1987',59X,'*',/,4X,'*',
     C65X,'*',/,4X,'*',15X,'ELECTRIC ENERGY RESEARCH',1X,'CENTER',
     C19X,'*',/,4X,'*',20X,'ELECTRICAL DEPARTMENT',24X,'*',/,4X,'*',
     C19X,'FACULTY OF ENGINEERINGS',23X,'*',/,4X,'*',18X,
     C'CHULALONGKORN UNIVERSITY',23X,'*',/,4X,'*',65X,'*',/,4X,
```

```
      C'*',12X,'LOW FREQUENCY OSCILATION ANALYSIS PROGRAM',12X,'*',/,
      C4X,'*',65X,'*',/,4X,'*',3X,'W. CHAROENSINLAPA',31X,':',2X,
      C'PROGRAMER',2X,'*',/,4X,'*',3X,'S. BHUMWUTTISARN',32X,':',2X,
      C'ADVISER',4X,'*',/,4X,'*',65X,'*',/,4X,67('*'),//)
       IK              =       8
       FMIN            =       0.0
       FINC            =       0.0
       MAX             =       0.0
       TOL             =       0.0
       MAXF            =       0
       WRITE(*,1)
     1 FORMAT(1H ,3X,'    CALCULATION SELECTION MODE',/,4X,
      *'(0) EIGENVALUE SEARCH',/,4X,
      *'(1) SWEEP FREQUENCY METHOD',//,4X,
      *'    ENTER YOUR SELECTED MODE > ',$)
       READ(*,2)ISEL
       ISEL            =       ISEL + 1
       IF(ISEL.GT.2)ISEL=2
     2 FORMAT(I5)
       WRITE(*,3)
     3 FORMAT(1H ,/,4X,'    PROGRAM SELECTION MODE',/,4X,
      *'(0) NORMAL MODE(READ DATA FORM OPENNED FILE)',/,4X,
      *'(1) INTERACTIVE MODE(READ DATA FORM KEYBOARD)',//,4X,
      *'    ENTER YOUR SELECTION MODE > ',$)
       READ(*,2)IC
C       WRITE(*,77)
    77 FORMAT(1H ,3X,'    OUTPUT SELECTION MODE',/,4X,
      *'(0) PRINT ON SCREEN',/,4X,
      *'(1) PRINT ON FILE(LFO.OP)',/,4X,
      *'    ENTER YOUR SELECTION MODE > ',$)
C       READ(*,2)IM
C       IP              =       1
C       IF(IM.EQ.1)IP=          6
       IP = 6
     4 CONTINUE
       CALL CONREAD(NB,NL,NT,NG,MAX,MAXF,IDRIVE,FREQCY,SBASE,TOL,WD,FO,
      *FMIN,FINC,IC,ISEL,IK,2)
       CALL LINREAD(SB,EB,ST,ET,RL,XS,RT,XT,TAP,PHSF,YSHUNT,
```

```
     *NB,NL,NT,IC,IK,2)
      CALL BUSREAD(ANG,VB,P,PL,Q,QL,TYPE,YCHARG,NB,IC,IK,2)
      CALL GENREAD(NG,GB,XD,XDP,XDPP,XQ,XQP,XQPP,TDOP,TDOPP,
     *TQOP,TQOPP,DAMP,XL,RA,HC,HBASE,TYPEX,IPSS,IGOV,XP,SAT1,SAT2,
     *IC,IK)
      CALL EXCREAD(NG,TYPEX,GA,TA,VRMAX,VRMIN,RV,XV,TA2,TA3,
     *XKE,TE,S1,S2,XKF,TF,TR,GP,GI,VBMAX,EFDMAX,EFDMIN,IC,IK,2)
      CALL PSSREAD(NG,IPSS,XG1,XT1,XT3,XT4,XT5,XT6,XT7,XT8,
     *VL1,VL2,IC,IK,2)
      CALL GOVREAD(NG,IGOV,G1,G2,G3,G4,T1,T2,T3,T4,T5,T6,IC,IK,2)
      WRITE(*,5)
    5 FORMAT(1H ,/,4X,'MODIFY ANY DATA ? (Y/N) > ',$)
      READ(*,6)OK
    6 FORMAT(A1)
      IF(OK.EQ.'Y')IK = 2
      IF(OK.EQ.'Y'.AND.IK.EQ.2)GO TO 4
      CALL WRITE
      CALL LINWRITE(SB,EB,ST,ET,RL,XS,RT,XT,TAP,PHSF,YSHUNT,NB,
     *NL,NT,IP)
      CALL BUSWRITE(ANG,VB,P,PL,Q,QL,TYPE,YCHARG,NB,IP)
      CALL GENWRITE(NG,HBASE,RA,XD,XQ,XDP,XQP,XDPP,
     *XQPP,XL,TDOP,TQOP,TDOPP,TQOPP,HC,DAMP,GB,IP)
      CALL EXCWRITE(NG,TYPEX,GA,TA,VRMAX,VRMIN,RV,XV,TA2,TA3,
     *XKE,TE,S1,S2,XKF,TF,TR,GP,GI,VBMAX,EFDMAX,EFDMIN,IP)
      CALL PSSWRITE(NG,IPSS,XG1,XT1,XT3,XT4,XT5,XT6,XT7,XT8,
     *VL1,VL2,IP)
      CALL GOVWRITE(NG,IGOV,G1,G2,G3,G4,T1,T2,T3,T4,T5,T6,IP)
      RETURN
      END
C     SUBROUTINE LINREAD
      SUBROUTINE LINREAD(SB,EB,ST,ET,RL,XL,RT,XT,TAP,PHSF,YSHUNT,
     *NB,NL,NT,IC,IK,N)
      INTEGER SB(1),EB(1),ST(1),ET(1)
      DIMENSION RL(1),XL(1),RT(1),XT(1),TAP(1),PHSF(1),YSHUNT(1)
      IF(IK.EQ.N)GO TO 6
      IF(IC.EQ.1)GO TO 6
      DO 3 I        =        1,NL
      READ(5,2)SB(I),EB(I),RL(I),XL(I),YSHUNT(I)
```

```
 2 FORMAT(2I5,3F10.4)
 3 CONTINUE
   DO 5 I       =       1,NT
   READ(5,4)ST(I),ET(I),RT(I),XT(I),TAP(I),PHSF(I)
 4 FORMAT(2I5,4F10.4)
 5 CONTINUE
   GO TO 23
 6 CONTINUE
   IF(IK.EQ.N)WRITE(*,50)
50 FORMAT(1H ,/,4X,
  *'     LINE DATA CORRECTION MENU',/,4X,
  *'( 0) QUIT',/,4X,'( 1) STARTING BUS',/,4X,'( 2) ENDING BUS',/,4X,
  *'( 3) LINE RESISTANCE',/,4X,'( 4) LINE REACTANCE',/,4X,
  *'( 5) CHARGING MVA',//,4X,'     ENTER YOUR SELECTED NUMBER > ',$)
   IF(IK.EQ.N)READ(*,8)ICR
   IF(IK.EQ.N.AND.ICR.LE.0)GO TO 70
   IF(IK.EQ.N)WRITE(*,51)
51 FORMAT(1H ,/,4X,'PLEASE SPECIFY THE INDEX OF THAT LINE > ',$)
   IF(IK.EQ.N)READ(*,8)I
   IF(IK.EQ.N)GO TO (53,54,55,56,57),ICR
   IF(IK.NE.N)I = 1
52 CONTINUE
 8 FORMAT(I5)
53 WRITE(*,10)I
10 FORMAT(1H ,/,4X,'LINE NO. ',I2,' STARTING FORM BUS NO. ',$)
   READ(*,8)SB(I)
   IF(IK.EQ.N)GO TO 16
54 WRITE(*,11)I
11 FORMAT(1H ,/,4X,'LINE NO. ',I2,' TERMINATED ON BUS NO. ',$)
   READ(*,8)EB(I)
   IF(IK.EQ.N)GO TO 16
55 WRITE(*,12)I
12 FORMAT(1H ,/,4X,'RESISTANCE IN PU. OF LINE NO. ',I2,
  *' = ',$)
   READ(*,15)RL(I)
   IF(IK.EQ.N)GO TO 16
56 WRITE(*,13)I
13 FORMAT(1H ,/,4X,'REACTANCE IN PU. OF LINE NO. ',I2,
```

```
     *' = ',$)
      READ(*,15)XL(I)
      IF(IK.EQ.N)GO TO 16
   57 WRITE(*,14)I
   14 FORMAT(1H ,/,4X,'CHARGING MVA OF LINE NO. ',I2,' = ',$)
      READ(*,15)YSHUNT(I)
      IF(IK.EQ.N)GO TO 16
      I              =        I + 1
      IF(I.LE.NL)GO TO 52
      GO TO 166
   15 FORMAT(F10.4)
   16 CONTINUE
      GO TO 6
  166 CONTINUE
      IF(IK.NE.N)I = 1
   70 CONTINUE
      IF(IK.EQ.N)WRITE(*,72)
   72 FORMAT(1H ,/,4X,'     TRANSFORMER DATA CORRECTION MENU',/,4X,
     *'( 0) QUIT',/,4X,
     *'( 1) PRIMARY SIDE BUS',/,4X,'( 2) SECONDARY SIDE BUS',/,4X,
     *'( 3) TRANSFORMER RESISTANCE',/,4X,
     *'( 4) TRANSFORMER REACTANCE',/,4X,'( 5) TAP SETTING',/,4X,
     *'(6 ) PHASE SHIFTING',//,4X,
     *'     ENTER YOUR SELECTED NUMBER > ',$)
      IF(IK.EQ.N)READ(*,8)ICR
      IF(IK.EQ.N.AND.ICR.LE.0)GO TO 22
      IF(IK.EQ.N)WRITE(*,73)
   73 FORMAT(1H1,3X,'PLEASE SPECIFY THE INDEX OF THAT TRANSFORMER > ',$)
      IF(IK.EQ.N)READ(*,8)I
      IF(IK.EQ.N)GO TO (74,75,76,77,78,79),ICR
   74 WRITE(*,17)I
   17 FORMAT(1H1,3X,'PRIMARY SIDE OF TRANSFORMER NO. ',I2,' = ',$)
      READ(*,8)ST(I)
      IF(IK.EQ.N)GO TO 70
   75 WRITE(*,18)I
   18 FORMAT(1H1,3X,'SECONDARY SIDE OF TRANSFORMER NO. ',I2,' = ',$)
      READ(*,8)ET(I)
      IF(IK.EQ.N)GO TO 70
```

```
   76 WRITE(*,19)I
   19 FORMAT(1H1,3X,'RESISTANCE IN PU. OF TRANSFORMER NO. ',I2,
     *' = ',$)
      READ(*,15)RT(I)
      IF(IK.EQ.N)GO TO 70
   77 WRITE(*,20)I
   20 FORMAT(1H1,3X,'REACTANCE IN PU. OF TRANSFORMER NO. ',I2,
     *' = ',$)
      READ(*,15)XT(I)
      IF(IK.EQ.N)GO TO 70
   78 WRITE(*,21)I
   21 FORMAT(1H1,3X,'TAP SETTING OF TRANSFORMER NO. ',I2,
     *' = ',$)
      READ(*,15)TAP(I)
      IF(IK.EQ.N)GO TO 70
   79 WRITE(*,80)I
   80 FORMAT(1H1,3X,'PHASE SHIFT IN DEGRE OF TRANSFORMER NO. ',I2,
     *' = ',$)
      READ(*,15)PHSF(I)
      IF(IK.EQ.N)GO TO 70
      I            =       I + 1
      IF(I.LE.NT)GO TO 70
   22 CONTINUE
   23 CONTINUE
      RETURN
      END
C     SUBROUTINE CONREAD
      SUBROUTINE CONREAD(NB,NL,NT,NG,MAX,MAXF,IDRIVE,FREQ,BASE,TOL,
     *WDRIF,FO,FMIN,FINC,IC,ISEL,IK,N)
      CHARACTER OK
      IF(IK.EQ.N)GO TO 3
      IF(IC.EQ.1)GO TO 3
      READ(5,1)NB,NL,NT,NG,MAX,IDRIVE,MAXF
    1 FORMAT(7I5)
      READ(5,2)FREQ,BASE,TOL,WDRIF,FO,FMIN,FINC
    2 FORMAT(8F10.4)
      GO TO 17
    3 CONTINUE
```

```
      IF(IK.EQ.N)WRITE(*,49)
   49 FORMAT(1H1,3X,
     *'   CORRECTION MENU NO. 1',/,4X,'( 0) QUIT',/,4X,
     *'( 1) TOTAL NUMBER OF BUSES',/,4X,
     *'( 2) TOTAL NUMBER OF LINES',/,4X,
     *'( 3) TOTAL NUMBER OF TRANSFORMERS',/,4X,
     *'( 4) TOTAL NUMBER OF GENERATORS',/,4X,
     *'( 5) FORCED GENERATOR',/,4X,
     *'( 6) FREQUENCY OF POWER SYSTEM',/,4X,
     *'( 7) BASE MVA OF SYSTEM',/,4X,
     *'( 8) PER UNIT SPEED DEVIATION OF FORCED GEN.',/,4X,
     *'( 9) MAXIMUM ITERATIONS FOR EIGENVALUE SEARCH',/,4X,
     *'(10) TOLERANCE FOR EIGENVALUE SEARCH',/,4X,
     *'(11) INITIAL INCREMENTAL FREQUENCY IN HERTZ',/,4X,
     *'(12) LOWEST FREQUENCY OF FREQUENCY BAND IN HERTZ',/,4X,
     *'      (FOR SWEEP FREQUENCY MODE)',/,4X,
     *'(13) FREQUENCY INCREMENT IN HERTZ',/,4X,
     *'      (FOR SWEEP FREQUENCY MODE)',/,4X,
     *'(14) FREQUENCY INCREMENT IN HERTZ',/,4X,
     *'      (FOR SWEEP FREQUENCY MODE)',/,4X,//,4X,
     *'      ENTER YOUR SELECTED NUMBER > ',$)
      IF(IK.EQ.N)READ(*,5)ICR
      IF(IK.EQ.N.AND.ICR.LE.0)GO TO 17
      IF(IK.EQ.N)GO TO (50,51,52,53,54,55,56,57,58,59,60,61,62,63),ICR
   50 WRITE(*,4)
    4 FORMAT(1H1,3X,'TOTAL NUMBER OF BUSES = ',$)
    5 FORMAT(I5)
      READ(*,5)NB
      IF(IK.EQ.N)GO TO 16
   51 WRITE(*,6)
    6 FORMAT(1H1,3X,'TOTAL NUMBER OF LINES = ',$)
      READ(*,5)NL
      IF(IK.EQ.N)GO TO 16
   52 WRITE(*,7)
    7 FORMAT(1H1,3X,'TOTAL NUMBER OF TRANSFORMERS = ',$)
      READ(*,5)NT
      IF(IK.EQ.N)GO TO 16
   53 WRITE(*,8)
```

```
 8 FORMAT(1H1,3X,'TOTAL NUMBER OF GENERATORS = ',$)
   READ(*,5)NG
   IF(IK.EQ.N)GO TO 16
54 WRITE(*,10)
10 FORMAT(1H1,3X,'FORCED SYNCHRONOUS MACHINE IS GEN. NO. = ',$)
   READ(*,5)IDRIVE
   IF(IK.EQ.N)GO TO 16
55 WRITE(*,11)
11 FORMAT(1H1,3X,'FREQUENCY OF POWER SYSTEM = ',$)
   READ(*,12)FREQ
   IF(IK.EQ.N)GO TO 16
12 FORMAT(F15.8)
56 WRITE(*,13)
13 FORMAT(1H1,3X,'BASE MVA OF SYSTEM = ',$)
   READ(*,12)BASE
   IF(IK.EQ.N)GO TO 16
57 WRITE(*,14)
14 FORMAT(1H1,3X,'PER UNIT SPEED DEVIATION OF FORCED GEN. = ',$)
   READ(*,12)WDRIF
   IF(IK.EQ.N)GO TO 16
   IF(ISEL.EQ.2)GO TO 34
58 WRITE(*,9)
 9 FORMAT(1H1,3X,'MAXIMUM ITERATIONS FOR EIGENVALUE SEARCH = ',$)
   READ(*,5)MAX
   IF(IK.EQ.N)GO TO 16
59 WRITE(*,15)
15 FORMAT(1H1,3X,'TOLERANCE FOR EIGENVALUE SEARCH = ',$)
   READ(*,12)TOL
   IF(IK.EQ.N)GO TO 16
60 WRITE(*,33)
33 FORMAT(1H1,3X,'INITIAL INCREMENTAL FREQUENCY IN HERTZ = ',$)
   READ(*,12)FO
   IF(IK.EQ.N)GO TO 16
   GO TO 17
34 CONTINUE
61 WRITE(*,35)
35 FORMAT(1H1,3X,'LOWEST FREQUENCY OF FREQUENCY BAND IN HERTZ = ',$)
   READ(*,12)FMIN
```

```
      IF(IK.EQ.N)GO TO 16
   62 WRITE(*,36)
   36 FORMAT(1H1,3X,'FREQUENCY INCREMENT IN HERTZ = ',$)
      READ(*,12)FINC
      IF(IK.EQ.N)GO TO 16
   63 WRITE(*,37)
   37 FORMAT(1H1,3X,'MAXIMUM POINTS OF FREQUENCY REQUIRED = ',$)
      READ(*,5)MAXF
      IF(IK.EQ.N)GO TO 16
      GO TO 17
   16 CONTINUE
      GO TO 3
   17 CONTINUE
      RETURN
      END
C     SUBROUTINE BUSREAD
      SUBROUTINE BUSREAD(ANG,VB,P,PL,Q,QL,TYPE,YCHARG,NB,IC,IK,N)
      DIMENSION P(1),PL(1),Q(1),QL(1),VB(1),ANG(1)
      INTEGER TYPE(1)
      COMPLEX YCHARG(1)
      IF(IK.EQ.N)GO TO 3
      IF(IC.EQ.1)GO TO 3
      DO 2 I        =       1,NB
      READ(5,1)P(I),PL(I),Q(I),QL(I),VB(I),ANG(I),YCH,TYPE(I)
    1 FORMAT(7F10.4,I5)
      YCHARG(I)     =       CMPLX(0.0,YCH)
    2 CONTINUE
      GO TO 15
    3 CONTINUE
      IF(IK.NE.N)I = 1
   50 CONTINUE
   72 FORMAT(I5)
      IF(IK.EQ.N)WRITE(*,77)
   77 FORMAT(1H1,3X,
     *'      BUS DATA CORRECTION MENU',/,4X,'( 0) QUIT',/,4X,
     *'( 1) REAL POWER GENERATION',/,4X,
     *'( 2) REACTIVE POWER GENERATION',/,4X,
     *'( 3) REAL POWER LOAD',/,4X,
```

```
     *'( 4) REACTIVE POWER LOAD',/,4X,
     *'( 5) MANITUDE OF VOLTAGE IN PU.',/,4X,
     *'( 6) PHASE ANGLE OF VOLTAGE IN DEGRE',/,4X,
     *'( 7) FIXED CAPACITANCE IN PU.',/,4X,
     *'( 8) BUS TYPE CODE',//,4X,
     *'      ENTER YOUR SELECTED NUMBER > ',$)
      IF(IK.EQ.N)READ(*,72)ICR
      IF(IK.EQ.N.AND.ICR.LE.0)GO TO 15
      IF(IK.EQ.N)WRITE(*,51)
   51 FORMAT(1H1,3X,'PLEASE SPECIFY THE INDEX OF THAT BUS > ',$)
      IF(IK.EQ.N)READ(*,72)I
      IF(IK.EQ.N)GO TO (52,53,54,55,56,57,58,59),ICR
   52 WRITE(*,4)I
    4 FORMAT(1H1,3X,'REAL POWER GENERATION IN MVA AT BUS NO. ',I2,
     *' = ',$)
      READ(*,5)P(I)
      IF(IK.EQ.N)GO TO 14
    5 FORMAT(F10.4)
   53 WRITE(*,6)I
    6 FORMAT(1H1,3X,'REACTIVE POWER GENERATION IN MVA AT BUS NO. ',I2,
     *' = ',$)
      READ(*,5)Q(I)
      IF(IK.EQ.N)GO TO 14
   54 WRITE(*,7)I
    7 FORMAT(1H1,3X,'REAL POWER LOAD IN MVA AT BUS NO. ',I2,' = ',$)
      READ(*,5)PL(I)
      IF(IK.EQ.N)GO TO 14
   55 WRITE(*,8)I
    8 FORMAT(1H1,3X,'REACTIVE POWER LOAD IN MVA AT BUS NO. ',I2,' = ',$)
      READ(*,5)QL(I)
      IF(IK.EQ.N)GO TO 14
   56 WRITE(*,9)I
    9 FORMAT(1H1,3X,'MAGNITUDE OF VOLTAGE IN PU. AT BUS NO. ',I2,
     *' = ',$)
      READ(*,5)VB(I)
      IF(IK.EQ.N)GO TO 14
   57   WRITE(*,10)I
   10 FORMAT(1H1,3X,'PHASE ANGLE OF VOLTAGE IN DEGRE AT BUS NO. ',I2,
```

```
     *' = ',$)
      READ(*,5)ANG(I)
      IF(IK.EQ.N)GO TO 14
   58 WRITE(*,11)I
   11 FORMAT(1H1,3X,'FIXED CAPACITANCE INPU. AT BUS NO. ',I2,
     *' = ',$)
      READ(*,5)YCH
      YCHARG(I)     =       CMPLX(0.0,YCH)
      IF(IK.EQ.N)GO TO 14
   59 WRITE(*,12)I
   12 FORMAT(1H1,3X,
     *'*****************************************************',/,4X,
     *'*                                                   *',/,4X,
     *'*                  BUS TYPE CODE                    *',/,4X,
     *'*            1   -    GENERATOR BUS                 *',/,4X,
     *'*            2   -    LOAD BUS                      *',/,4X,
     *'*            3   -    INFINITE BUS                  *',/,4X,
     *'*                                                   *',/,4X,
     *'*****************************************************',////,4X,
     *'TYPE OF BUS NO. ',I2,' = ',$)
      READ(*,13)TYPE(I)
   13 FORMAT(I5)
      IF(IK.EQ.N)GO TO 14
      I             =       I + 1
      IF(I.LE.NB)GO TO 50
      GO TO 15
   14 CONTINUE
      GO TO 50
   15 CONTINUE
      RETURN
      END
C     SUBROUTINE GOVREAD
      SUBROUTINE GOVREAD(NG,IGOV,G1,G2,G3,G4,T1,T2,T3,T4,T5,T6,IC,IK,N)
      INTEGER IGOV(1)
      DIMENSION G1(1),G2(1),G3(1),G4(1),T1(1),T2(1),T3(1),T4(1),T5(1),
     *T6(1)
      IF(IK.EQ.N)GO TO 9
      IX            =       0
```

```
      DO 2 I        =       1,NG
      IF(IGOV(I).EQ.0)GO TO 1
      IX            =       IX + 1
    1 CONTINUE
    2 CONTINUE
      IF(IX.EQ.0)GO TO 28
      IF(IC.EQ.1)GO TO 9
      DO 6 I        =       1,NG
      IF(IGOV(I).NE.1)GO TO 5
      READ(5,4)G1(I),T1(I),T2(I),T3(I)
    4 FORMAT(8F10.4)
    5 CONTINUE
    6 CONTINUE
      DO 8 I        =       1,NG
      IF(IGOV(I).NE.1)GO TO 7
      READ(5,4)G2(I),T4(I),T5(I),G3(I),G4(I),T6(I)
    7 CONTINUE
    8 CONTINUE
      GO TO 23
    9 CONTINUE
      IF(IK.NE.N)I=1
   50 CONTINUE
      IF(IK.EQ.N)WRITE(*,51)
   51 FORMAT(1H1,3X,'     SPEED-GOVERNING SYSTEM DATA CORRECTION MENU',
     */,4X,'( 0) QUIT',/,4X,
     *'( 1) K1',/,4X,'( 2) K2',/,4X,'( 3) K3',/,4X,'( 4) K4',/,4X,
     *'( 5) T1',/,4X,'( 6) T2',/,4X,'( 7) T3',/,4X,'( 8) T4',/,4X,
     *'( 9) T5',/,4X,'(10) T6',//,4X,
     *'     ENTER YOUR SELECTED NUMBER > ',$)
      IF(IK.EQ.N)READ(*,72)ICR
      IF(IK.EQ.N.AND.ICR.LE.0)GO TO 28
      IF(IK.EQ.N)WRITE(*,49)
   49 FORMAT(1H1,3X,'PLEASE SPECIFY THE INDEX OF THAT SPEED',
     *' GOVERNER > ',$)
      IF(IK.EQ.N)READ(*,72)I
   72 FORMAT(I5)
      IF(IK.EQ.N)GO TO (52,53,54,55,56,57,58,59,60,61),ICR
      IF(IGOV(I).NE.1)GO TO 22
```

```
52 WRITE(*,10)I
10 FORMAT(1H1,3X,'K1 OF SPEED GOVERNER NO. ',I2,' = ',$)
   READ(*,11)G1(I)
   IF(IK.EQ.N)GO TO 21
11 FORMAT(F10.4)
53 WRITE(*,12)I
12 FORMAT(1H1,3X,'K2 OF SPEED GOVERNER NO. ',I2,' = ',$)
   READ(*,11)G2(I)
   IF(IK.EQ.N)GO TO 21
54 WRITE(*,13)I
13 FORMAT(1H1,3X,'K3 OF SPEED GOVERNER NO. ',I2,' = ',$)
   READ(*,11)G3(I)
   IF(IK.EQ.N)GO TO 21
55 WRITE(*,14)I
14 FORMAT(1H1,3X,'K4 OF SPEED GOVERNER NO. ',I2,' = ',$)
   READ(*,11)G4(I)
   IF(IK.EQ.N)GO TO 21
56 WRITE(*,15)I
15 FORMAT(1H1,3X,'T1 OF SPEED GOVERNER NO. ',I2,' = ',$)
   READ(*,11)T1(I)
   IF(IK.EQ.N)GO TO 21
57 WRITE(*,16)I
16 FORMAT(1H1,3X,'T2 OF SPEED GOVERNER NO. ',I2,' = ',$)
   READ(*,11)T2(I)
   IF(IK.EQ.N)GO TO 21
58 WRITE(*,17)I
17 FORMAT(1H1,3X,'T3 OF SPEED GOVERNER NO. ',I2,' = ',$)
   READ(*,11)T3(I)
   IF(IK.EQ.N)GO TO 21
59 WRITE(*,18)I
18 FORMAT(1H1,3X,'T4 OF SPEED GOVERNER NO. ',I2,' = ',$)
   READ(*,11)T4(I)
   IF(IK.EQ.N)GO TO 21
60 WRITE(*,19)I
19 FORMAT(1H1,3X,'T5 OF SPEED GOVERNER NO. ',I2,' = ',$)
   READ(*,11)T5(I)
   IF(IK.EQ.N)GO TO 21
61 WRITE(*,20)I
```

```
   20 FORMAT(1H1,3X,'T6 OF SPEED GOVERNER NO. ',I2,' = ',$)
      READ(*,11)T6(I)
      IF(IK.EQ.N)GO TO 21
      GO TO 22
   21 CONTINUE
      GO TO 50
   22 CONTINUE
      I            =       I + 1
      IF(I.LE.NG)GO TO 50
   23 CONTINUE
   28 CONTINUE
      RETURN
      END
C     SUBROUTINE PSSREAD
      SUBROUTINE PSSREAD(NG,IPSS,XG1,XT1,XT3,XT4,XT5,XT6,XT7,XT8,
     *VL1,VL2,IC,IK,N)
      DIMENSION XG1(1),XT1(1),XT3(1),XT4(1),XT5(1),XT6(1),XT7(1),
     *XT8(1),VL1(1),VL2(1)
      INTEGER IPSS(1)
      IF(IK.EQ.N)GO TO 6
      IX           =       0
      DO 111 I     =       1,NG
      IF(IPSS(I).EQ.0)GO TO 111
      IX           =       IX + 1
  111 CONTINUE
      IF(IX.EQ.0)GO TO 20
      IF(IC.EQ.1)GO TO 6
      DO 3 I       =       1,NG
      IF(IPSS(I).NE.1)GO TO 2
      READ(5,1)XG1(I),XT1(I),XT3(I),XT4(I),XT5(I),XT6(I),XT7(I),
     *XT8(I)
    1 FORMAT(8F10.4)
    2 CONTINUE
    3 CONTINUE
      IF(IPSS(I).NE.1)GO TO 4
      READ(5,1)VL1(I),VL2(I)
    4 CONTINUE
    5 CONTINUE
```

```
      GO TO 20
    6 CONTINUE
      IF(IK.NE.N)I=1
   50 CONTINUE
   52 FORMAT(I5)
   53 CONTINUE
      IF(IK.EQ.N)WRITE(*,54)
   54 FORMAT(1H1,3X,
     *'      POWER SYSTEM STABILIZER DATA CORRECTION MENU',/,4X,
     *'( 0) QUIT',/,4X,
     *'( 1) K1',/,4X,'( 2) T1',/,4X,'( 3) T3',/,4X,'( 4) T4',/,4X,
     *'( 5) T5',/,4X,'( 6) T6',/,4X,'( 7) T7',/,4X,'( 8) T8',/,4X,
     *'( 9) L1',/,4X,'(10) L2',//,4X,'ENTER YOUR SELECTED NUMBER > ',$)
      IF(IK.EQ.N)READ(*,52)ICR
      IF(IK.EQ.N.AND.ICR.LE.0)GO TO 20
      IF(IK.EQ.N)WRITE(*,51)
   51 FORMAT(1H1,3X,'PLEASE SPECIFY THE INDEX OF THAT PSS > ',$)
      IF(IK.EQ.N)READ(*,52)I
      IF(IK.EQ.N)GO TO (55,56,57,58,60,61,62,63,64,65),ICR
      IF(IPSS(I).NE.1)GO TO 18
   55 WRITE(*,7)I
    7 FORMAT(1H1,3X,'K1 OF POWER SYSTEM STABILIZER NO. ',I2,' = ',$)
      READ(*,8)XG1(I)
      IF(IK.EQ.N)GO TO 19
    8 FORMAT(F10.4)
   56 WRITE(*,9)I
    9 FORMAT(1H1,3X,'T1 OF POWER SYSTEM STABILIZER NO. ',I2,' = ',$)
      READ(*,8)XT1(I)
      IF(IK.EQ.N)GO TO 19
   57 WRITE(*,10)I
   10 FORMAT(1H1,3X,'T3 OF POWER SYSTEM STABILIZER NO. ',I2,' = ',$)
      READ(*,8)XT3(I)
      IF(IK.EQ.N)GO TO 19
   58 WRITE(*,11)I
   11 FORMAT(1H1,3X,'T4 OF POWER SYSTEM STABILIZER NO. ',I2,' = ',$)
      READ(*,8)XT4(I)
      IF(IK.EQ.N)GO TO 19
   60 WRITE(*,12)I
```

```
 12 FORMAT(1H1,3X,'T5 OF POWER SYSTEM STABILIZER NO. ',I2,' = ',$)
    READ(*,8)XT5(I)
    IF(IK.EQ.N)GO TO 19
 61 WRITE(*,13)I
 13 FORMAT(1H1,3X,'T6 OF POWER SYSTEM STABILIZER NO. ',I2,' = ',$)
    READ(*,8)XT6(I)
    IF(IK.EQ.N)GO TO 19
 62 WRITE(*,14)I
 14 FORMAT(1H1,3X,'T7 OF POWER SYSTEM STABILIZER NO. ',I2,' = ',$)
    READ(*,8)XT7(I)
    IF(IK.EQ.N)GO TO 19
 63 WRITE(*,15)I
 15 FORMAT(1H1,3X,'T8 OF POWER SYSTEM STABILIZER NO. ',I2,' = ',$)
    READ(*,8)XT8(I)
    IF(IK.EQ.N)GO TO 19
 64 WRITE(*,16)I
 16 FORMAT(1H1,3X,'L1 OF POWER SYSTEM STABILIZER NO. ',I2,' = ',$)
    READ(*,8)VL1(I)
    IF(IK.EQ.N)GO TO 19
 65 WRITE(*,17)I
 17 FORMAT(1H1,3X,'L2 OF POWER SYSTEM STABILIZER NO. ',I2,' = ',$)
    READ(*,8)VL2(I)
    IF(IK.EQ.N)GO TO 19
 18 CONTINUE
    I            =       I + 1
    IF(I.LE.NG)GO TO 50
    GO TO 20
 19 CONTINUE
    GO TO 50
 20 CONTINUE
    RETURN
    END
C    SUBROUTINE EXCREAD
     SUBROUTINE EXCREAD(NG,TYPEX,GA,TA,VRMAX,VRMIN,RV,XV,TA2,TA3,
    *XKE,TE,S1,S2,XKF,TF,TR,GP,GI,VBMAX,EFDMAX,EFDMIN,IC,IK,N)
     DIMENSION GA(1),TA(1),VRMAX(1),VRMIN(1),RV(1),XV(1),TA2(1),TA3(1),
    *XKE(1),TE(1),S1(1),S2(1),XKF(1),TF(1),TR(1),GP(1),GI(1),VBMAX(1),
    *EFDMAX(1),EFDMIN(1)
```

```
      INTEGER TYPEX(1)
      IF(IK.EQ.N)GO TO 6
      IF(IC.EQ.1)GO TO 6
      DO 2 I        =        1,NG
      IF(TYPEX(I).EQ.0)GO TO 2
      READ(5,1)GA(I),TA(I),VRMAX(I),VRMIN(I),RV(I),XV(I),TA2(I),TA3(I)
    1 FORMAT(8F10.4)
    2 CONTINUE
      DO 5 I        =        1,NG
      IF(TYPEX(I).EQ.0)GO TO 4
      IF(TYPEX(I).EQ.3)GO TO 3
      READ(5,1)XKE(I),TE(I),S1(I),S2(I),XKF(I),TF(I),TR(I)
      GO TO 4
    3 READ(5,1)XKE(I),TE(I),GP(I),GI(I),VBMAX(I),XKF(I),TF(I),TR(I)
    4 CONTINUE
    5 CONTINUE
      DO 300 I      =        1,NG
      IF(TYPEX(I).EQ.0)GO TO 299
      READ(5,1)EFDMAX(I),EFDMIN(I)
  299 CONTINUE
  300 CONTINUE
      GO TO 35
    6 CONTINUE
   51 FORMAT(A1)
   52 CONTINUE
   54 CONTINUE
   88 IF(IK.EQ.N)WRITE(*,55)
   55 FORMAT(1H1,3X,'     EXCITER DATA CORRECTION MENU',/,4X,
     *'( 0) QUIT',/,4X,
     *'( 1) TA',/,4X,'( 2) TB',/,4X,'( 3) TC',/,4X,'( 4) TE',/,4X,
     *'( 5) TF',/,4X,'( 6) KA',/,4X,'( 7) KE',/,4X,'( 8) KF',/,4X,
     *'( 9) SATURATION FUNCTION AT 1.0 PU. VOLTAGE',/,4X,
     *'(10) SATURATION FUNCTION AT 1.2 PU. VOLTAGE',/,4X,
     *'(11) VRMAX',/,4X,'(12) VRMIN',/,4X,'(13) KP',/,4X,'(14) KI',/,4X,
     *'(15) VBMAX',/,4X,'(16) TRH',/,4X,'(17) TR',/,4X,
     *'(18) COMPENSATING RESISTANCE',/,4X,'(19) COMPENSATING REACTANCE',
     */,4X,'(20) UPPER LIMIT OF EFD(EFDMAX)',/,4X,
     *'(21) LOWER LIMIT OF EFD(EFDMIN)',//,4X,
```

```
     *'      ENTER YOUR SELECTED NUMBER > ',$)
      IF(IK.EQ.N)READ(*,888)ICR
      IF(IK.EQ.N.AND.ICR.LE.0)GO TO 35
      IF(IK.EQ.N)WRITE(*,53)
   53 FORMAT(1H1,3X,'PLEASE SPECIFY THE INDEX OF THAT EXCITER > ',$)
      IF(IK.EQ.N)READ(*,888)I
  888 FORMAT(I10)
      IF(IK.EQ.N)GO TO (60,61,62,64,66,59,63,65,100,67,57,58,
     *68,69,70,71,74,72,73,200,202),ICR
      IF(IK.NE.N)I=1
   56 CONTINUE
      IF(TYPEX(I).EQ.0)GO TO 34
   57 WRITE(*,7)I
    7 FORMAT(1H1,3X,'VRMAX OF EXCITER NO. ',I2,' = ',$)
      READ(*,8)VRMAX(I)
    8 FORMAT(F20.10)
      IF(IK.EQ.N)GO TO 33
   58 WRITE(*,9)I
    9 FORMAT(1H1,3X,'VRMIN OF EXCITER NO. ',I2,' = ',$)
      READ(*,8)VRMIN(I)
      IF(IK.EQ.N)GO TO 33
      IF(TYPEX(I).EQ.4)GO TO 14
   59 WRITE(*,10)I
   10 FORMAT(1H1,3X,'KA OF EXCITER NO. ',I2,' = ',$)
      READ(*,8)GA(I)
      IF(IK.EQ.N)GO TO 33
   60 WRITE(*,11)I
   11 FORMAT(1H1,3X,'TA OF EXCITER NO. ',I2,' = ',$)
      READ(*,8)TA(I)
      IF(IK.EQ.N)GO TO 33
      IF(TYPEX(I).EQ.3)GO TO 16
   61 WRITE(*,12)I
   12 FORMAT(1H1,3X,'TB OF EXCITER NO. ',I2,' = ',$)
      READ(*,8)TA2(I)
      IF(IK.EQ.N)GO TO 33
   62 WRITE(*,13)I
   13 FORMAT(1H1,3X,'TC OF EXCITER NO. ',I2,' = ',$)
      READ(*,8)TA3(I)
```

```
      IF(IK.EQ.N)GO TO 33
   14 CONTINUE
      IF(TYPEX(I).EQ.6)GO TO 18
   63 WRITE(*,15)I
   15 FORMAT(1H1,3X,'KE OF EXCITER NO. ',I2,' = ',$)
      READ(*,8)XKE(I)
   16 CONTINUE
      IF(IK.EQ.N)GO TO 33
   64 WRITE(*,17)I
   17 FORMAT(1H1,3X,'TE OF EXCITER NO. ',I2,' = ',$)
      READ(*,8)TE(I)
      IF(IK.EQ.N)GO TO 33
      IF(TYPEX(I).EQ.4)GO TO 100
   18 CONTINUE
   65 WRITE(*,19)I
   19 FORMAT(1H1,3X,'KF OF EXCITER NO. ',I2,' = ',$)
      READ(*,8)XKF(I)
      IF(IK.EQ.N)GO TO 33
   66 WRITE(*,20)I
   20 FORMAT(1H1,3X,'TF OF EXCITER NO. ',I2,' = ',$)
      READ(*,8)TF(I)
      IF(IK.EQ.N)GO TO 33
      IF(TYPEX(I).EQ.3.OR.TYPEX(I).EQ.6)GO TO 24
  100 WRITE(*,21)I
   21 FORMAT(1H1,3X,'SATURATON FUINCTION AT 1.0 PU.'/,4X,
     *'OF EXCITER NO. ',I2,' = ',$)
      READ(*,8)S1(I)
      IF(IK.EQ.N)GO TO 33
   67 WRITE(*,22)I
   22 FORMAT(1H1,3X,'SATURATON FUINCTION AT 1.2 PU.'/,4X,
     *'OF EXCITER NO. ',I2,' = ',$)
      READ(*,8)S2(I)
      IF(IK.EQ.N)GO TO 33
      IF(TYPEX(I).EQ.4)GO TO 27
   24 CONTINUE
      IF(TYPEX(I).NE.3)GO TO 27
   68 WRITE(*,25)I
   25 FORMAT(1H1,3X,'KP OF EXCITER NO. ',I2,' = ',$)
```

```
      READ(*,8)GP(I)
      IF(IK.EQ.N)GO TO 33
   69 WRITE(*,26)I
   26 FORMAT(1H1,3X,'KI OF EXCITER NO. ',I2,' = ',$)
      READ(*,8)GI(I)
      IF(IK.EQ.N)GO TO 33
   70 WRITE(*,111)I
  111 FORMAT(1H1,3X,'VBMAX OF EXCITER NO. ',I2,' = ',$)
      READ(*,8)VBMAX(I)
      IF(IK.EQ.N)GO TO 33
   27 CONTINUE
      IF(TYPEX(I).NE.4)GO TO 29
   71 WRITE(*,28)I
   28 FORMAT(1H1,3X,'TRH OF EXCITER NO. ',I2,' = ',$)
      READ(*,8)TA(I)
      IF(IK.EQ.N)GO TO 33
   29 CONTINUE
   72 WRITE(*,30)I
   30 FORMAT(1H1,3X,'COMPENSATION RESISTANCE OF EXCITER NO. ',I2,
     *' = ',$)
      READ(*,8)RV(I)
      IF(IK.EQ.N)GO TO 33
   73 WRITE(*,31)I
   31 FORMAT(1H1,3X,'COMPENSATION REACTANCE OF EXCITER NO. ',I2,' = ',$)
      READ(*,8)XV(I)
      IF(IK.EQ.N)GO TO 33
   74 WRITE(*,32)I
   32 FORMAT(1H1,3X,'TR OF EXCITER NO. ',I2,' = ',$)
      READ(*,8)TR(I)
      IF(IK.EQ.N)GO TO 33
  200 WRITE(*,201)I
  201 FORMAT(1H1,3X,'UPPER LIMIT OF EFD OF EXCITER NO. ',I2,' = ',$)
      READ(*,8)EFDMAX(I)
      IF(IK.EQ.N)GO TO 33
  202 WRITE(*,203)I
  203 FORMAT(1H1,3X,'LOWER LIMIT OF EFD OF EXCITER NO. ',I2,' = ',$)
      READ(*,8)EFDMIN(I)
      IF(IK.EQ.N)GO TO 33
```

```
      GO TO 34
   33 CONTINUE
      GO TO 52
   34 CONTINUE
      I            =       I + 1
      IF(I.LE.NG)GO TO 56
   35 CONTINUE
      RETURN
      END
C     SUBROUTINE LINWRITE
      SUBROUTINE LINWRITE(SB,EB,ST,ET,RL,XL,RT,XT,TAP,PHSF,YSHUNT,
     *NB,NL,NT,IP)
      INTEGER SB(1),EB(1),ST(1),ET(1)
      DIMENSION RL(1),XL(1),RT(1),XT(1),TAP(1),YSHUNT(1),PHSF(1)
      WRITE(IP,1)
    1 FORMAT(1H ,3X,60('-'),/,4X,
     *'I      T R A N S M I S S I O N   L I N E S   D A T A     I',
     */,4X,60('-'),/,4X,
     *'I  BUS  I  BUS  I  R(PU)  I  X(PU)  I  CHARGING  I',
     */,4X,60('-'))
    2 FORMAT(1H ,3X,'I',3X,I2,4X,'I',3X,I2,4X,'I ',F7.2,2X,' I ',F7.2,
     *2X,' I ',F9.2,3X,' I',/,4X,60('-'))
      DO 3 I       =       1,NL
      WRITE(IP,2)SB(I),EB(I),RL(I),XL(I),YSHUNT(I)
    3 CONTINUE
      WRITE(IP,4)
    4 FORMAT(1H ,3X,72('-'),/,4X,'I',18X,'T R A N S F O R M E R S',
     *5X,'D A T A',17X,'I',/,4X,72('-'),/,4X,'I  PRI  I  SEC  ',
     *'I  R(PU)  I  X(PU)  I     TAP     I',
     *'  PH.SHIFT I',/,4X,72('-'))
    5 FORMAT(1H ,3X,'I',3X,I2,4X,'I',3X,I2,4X,'I ',F7.2,2X,' I ',F7.2,
     *2X,' I ',F9.2,3X,' I',3X,F5.2,3X,'I',/,4X,72('-'))
      DO 6 I       =       1,NT
      WRITE(IP,5)ST(I),ET(I),RT(I),XT(I),TAP(I),PHSF(I)
    6 CONTINUE
      RETURN
      END
C     SUBROUTINE GENREAD
```

```
      SUBROUTINE GENREAD(NG,GB,XD,XDP,XDPP,XQ,XQP,XQPP,TDOP,TDOPP,
     *TQOP,TQOPP,DAMP,XL,RA,HC,HBASE,TYPEX,IPSS,IGOV,XP,SAT1,SAT2,IC,
     *IK)
      DIMENSION DAMP(1),HC(1),HBASE(1),RA(1),SAT1(1),SAT2(1),TDOP(1),
     *TDOPP(1),TQOP(1),TQOPP(1),XD(1),XDP(1),XDPP(1),XL(1),XP(1),XQ(1),
     *XQP(1),XQPP(1)
      INTEGER GB(1),TYPEX(1),IPSS(1),IGOV(1)
      IOK             =       IK + 1
      IF(IOK.EQ.3)GO TO 8
      IF(IC.EQ.1)GO TO 8
    1 FORMAT(I5)
      DO 3 I          =       1,NG
      READ(5,2)RA(I),XL(I),XD(I),XDP(I),XDPP(I),XQ(I),XQP(I),XQPP(I)
    2 FORMAT(8F10.4)
    3 CONTINUE
      DO 5 I          =       1,NG
      READ(5,2)TDOP(I),TDOPP(I),TQOP(I),TQOPP(I),HC(I),HBASE(I),DAMP(I)
    5 CONTINUE
      DO 7 I          =       1,NG
      READ(5,6)XP(I),SAT1(I),SAT2(I),TYPEX(I),IPSS(I),IGOV(I),GB(I)
    6 FORMAT(3F10.4,4I5)
    7 CONTINUE
      GO TO 92
    8 CONTINUE
   10 FORMAT(I5)
      IF(IOK.NE.3)I = 1
   88 CONTINUE
      IF(IOK.EQ.3)WRITE(*,50)
   50 FORMAT(1H1,3X,'GENERATOR DATA CORRECTION MENU',/,4X,
     *'( 0) QUIT',/,4X,'( 1) D - AXIS SYNCHRONOUS REACTANCE',/,4X,
     *'( 2) D - AXIS TRANSIENT REACTANCE',/,4X,'( 3) D - AXIS S',
     *'UBTRANSIENT REACTANCE',/,4X,'( 4) Q - AXIS SYNCHRONOUS R',
     *'EACTANCE',/,4X,'( 5) Q - AXIS TRANSIENT REACTANCE',/,4X,
     *'( 6) Q - AXIS SUBTRANSIENT REACTANCE',/,4X,'( 7) LEAKAGE',
     *' REACTANCE',/,4X,'( 8) ARMATURE RESISTANCE',/,4X,
     *'( 9) POINTER REACTANCE',/,4X,'(10) SATURATION FUNCTION A',
     *'T 1.0 PU. VOLTAGE',/,4X,'(11) SATURATION FUNCTION AT 1.2',
     *' PU. VOLTAGE',/,4X,'(12) INERTIA CONSTANCE',/,4X,
```

```
     *'(13) MVA RATING',/,4X,'(14) DAMPING COEFFICIENT',/,4X,
     *'(15) D - AXIS OPENNED CIRCUIT TRANSIENT TIME CONSTANCE',/,4X,
     *'(16) D - AXIS OPENNED CIRCUIT SUBTRANSIENT TIME CONSTANCE',/,4X,
     *'(17) Q - AXIS OPENNED CIRCUIT TRANSIENT TIME CONSTANCE',/,4X,
     *'(18) Q - AXIS OPENNED CIRCUIT SUBTRANSIENT TIME CONSTANCE',/,4X,
     *'(19) TERMINAL BUS OF GENERATOR',/,4X,'(20) EXCITER TYPE CODE',
     */,4X,'(21) POWER SYSTEM STABILIZER CODE',/,4X,'(22) SPEED GO',
     *'VERNING SYSTEM CODE',/,4X,'ENTER YOUR SELECTED NUMBER > ',$)
      IF(IOK.EQ.3)READ(*,27)ICR
      IF(IOK.EQ.3.AND.ICR.LE.0)GO TO 92
      IF(IOK.EQ.3)WRITE(*,51)
   51 FORMAT(1H1,3X,'PLEASE SPECIFY THE INDEX OF THAT GENERATOR > ',$)
      IF(IOK.EQ.3)READ(*,27)I
      IF(IOK.EQ.3)GO TO (52,53,54,55,56,57,58,59,60,61,62,63,64,65,
     *66,67,68,69,70,71,72,73),ICR
   52 WRITE(*,11)I
   11 FORMAT(1H1,3X,'D - AXIS SYNCHRONOUS REACTANCE OF GEN. NO. ',I2,
     *' = ',$)
      READ(*,12)XD(I)
      IF(IOK.EQ.3)GO TO 74
   12 FORMAT(F10.4)
   53 WRITE(*,13)I
   13 FORMAT(1H1,3X,'D - AXIS TRANSIENT REACTANCE OF GEN. NO. ',I2,
     *' = ',$)
      READ(*,12)XDP(I)
      IF(IOK.EQ.3)GO TO 74
   54 WRITE(*,14)I
   14 FORMAT(1H1,3X,'D - AXIS SUBTRANSIENT REACTANCE OF GEN. NO. ',I2,
     *' = ',$)
      READ(*,12)XDPP(I)
      IF(IOK.EQ.3)GO TO 74
   55 WRITE(*,15)I
   15 FORMAT(1H1,3X,'Q - AXIS SYNCHRONOUS REACTANCE OF GEN. NO. ',I2,
     *' = ',$)
      READ(*,12)XQ(I)
      IF(IOK.EQ.3)GO TO 74
   56 WRITE(*,16)I
   16 FORMAT(1H1,3X,'Q - AXIS TRANSIENT REACTANCE OF GEN. NO. ',I2,
```

```
     *' = ',$)
      READ(*,12)XQP(I)
      IF(IOK.EQ.3)GO TO 74
   57 WRITE(*,17)I
   17 FORMAT(1H1,3X,'Q - AXIS SUBTRANSIENT REACTANCE OF GEN. NO. ',I2,
     *' = ',$)
      READ(*,12)XQPP(I)
      IF(IOK.EQ.3)GO TO 74
   58 WRITE(*,18)I
   18 FORMAT(1H1,3X,'LEAKAGE REACTANCE OF GEN. NO. ',I2,' = ',$)
      READ(*,12)XL(I)
      IF(IOK.EQ.3)GO TO 74
   59 WRITE(*,19)I
   19 FORMAT(1H1,3X,'ARMATURE RESISTANCE OF GEN. NO. ',I2,' = ',$)
      READ(*,12)RA(I)
      IF(IOK.EQ.3)GO TO 74
   60 WRITE(*,20)I
   20 FORMAT(1H1,3X,'POINTER REACTANCE OF GEN. NO. ',I2,' = ')
      READ(*,12)XP(I)
      IF(IOK.EQ.3)GO TO 74
   61 WRITE(*,21)I
   21 FORMAT(1H1,3X,'SATURATION FUNCTION AT 1.0 PU. VOLTAGE',/,4X,
     *'OF GEN. NO. ',I2,' = ')
      READ(*,12)SAT1(I)
      IF(IOK.EQ.3)GO TO 74
   62 WRITE(*,22)I
   22 FORMAT(1H1,3X,'SATURATION FUNCTION AT 1.2 PU. VOLTAGE',/,4X,
     *'OF GEN. NO. ',I2,' = ')
      READ(*,12)SAT2(I)
      IF(IOK.EQ.3)GO TO 74
   63 WRITE(*,23)I
   23 FORMAT(1H1,3X,'INERTIA CONSTANT OF GEN. NO. ',I2,' = ')
      READ(*,12)HC(I)
      IF(IOK.EQ.3)GO TO 74
   64 WRITE(*,24)I
   24 FORMAT(1H1,3X,'MVA RATING OF GEN. NO. ',I2,' = ')
      READ(*,12)HBASE(I)
      IF(IOK.EQ.3)GO TO 74
```

```
 65 WRITE(*,25)I
 25 FORMAT(1H1,3X,'DAMPING COEFFICIENT OF GEN. NO. ',I2,' = ')
    READ(*,12)DAMP(I)
    IF(IOK.EQ.3)GO TO 74
 66 WRITE(*,35)I
 35 FORMAT(1H1,3X,'D - AXIS OPENNED CIRCUIT TRANSIENT',/,4X,
   *'TIME CONSTANT OF GEN. NO. ',I2,' = ')
    READ(*,12)TDOP(I)
    IF(IOK.EQ.3)GO TO 74
 67 WRITE(*,36)I
 36 FORMAT(1H1,3X,'D - AXIS OPENNED CIRCUIT SUBTRANSIENT',/,4X,
   *'TIME CONSTANT OF GEN. NO. ',I2,' = ')
    READ(*,12)TDOPP(I)
    IF(IOK.EQ.3)GO TO 74
 68 WRITE(*,37)I
 37 FORMAT(1H1,3X,'Q - AXIS OPENNED CIRCUIT TRANSIENT',/,4X,
   *'TIME CONSTANT OF GEN. NO. ',I2,' = ')
    READ(*,12)TQOP(I)
    IF(IOK.EQ.3)GO TO 74
 69 WRITE(*,38)I
 38 FORMAT(1H1,3X,'Q - AXIS OPENNED CIRCUIT SUBTRANSIENT',/,4X,
   *'TIME CONSTANT OF GEN. NO. ',I2,' = ')
    READ(*,12)TQOPP(I)
    IF(IOK.EQ.3)GO TO 74
 70 WRITE(*,26)I
 26 FORMAT(1H1,3X,'GEN. NO. ',I2,' CONNECTED TO BUS NO. = ')
    READ(*,27)GB(I)
    IF(IOK.EQ.3)GO TO 74
 27 FORMAT(I5)
 71 WRITE(*,28)I
 28 FORMAT(1H1,3X,'EXCITER AT GEN. NO. ',I2,' TYPE = ')
    READ(*,27)TYPEX(I)
    IF(IOK.EQ.3)GO TO 74
 72 WRITE(*,100)I
100 FORMAT(1H1,3X,'IF THERE IS A POWER SYSTEM STABILIZER',/,4X,
   *'AT GEN. NO. ',I2,' PLEASE KEY "1"')
    READ(*,27)IPSS(I)
    IF(IOK.EQ.3)GO TO 74
```

```
 73 WRITE(*,101)I
101 FORMAT(1H1,3X,'IF THERE IS A SPEED-GOVERNING SYSTEM',/,4X,
   *'AT GEN. NO. ',I2,' PLEASE KEY "1"')
    READ(*,27)IGOV(I)
    IF(IOK.EQ.3)GO TO 74
 29 CONTINUE
    I           =       I + 1
    IF(I.LE.NG)GO TO 88
    GO TO 92
 30 CONTINUE
 74 CONTINUE
    GO TO 88
 92 CONTINUE
    RETURN
    END
C     SUBROUTINE EXCWRITE
    SUBROUTINE EXCWRITE(NG,TYPEX,GA,TA,VRMAX,VRMIN,RC,XC,TA2,TA3,
   *XKE,TE,S1,S2,XKF,TF,TR,GP,GI,VBMAX,EFDMAX,EFDMIN,IP)
    DIMENSION GA(1),TA(1),VRMAX(1),VRMIN(1),RC(1),XC(1),TA2(1),TA3(1),
   *XKE(1),TE(1),S1(1),S2(1),XKF(1),TF(1),TR(1),GP(1),GI(1),VBMAX(1),
   *EFDMAX(1),EFDMIN(1)
    INTEGER TYPEX(1)
    IJ          =       0
    DO 100 I    =       1,NG
    IF(TYPEX(I).EQ.0)GO TO 99
    IJ          =       IJ + 1
 99 CONTINUE
100 CONTINUE
    IF(IJ.EQ.0)GO TO 11
    WRITE(IP,1)
  1 FORMAT(1H ,3X,60('-'),/,4X,
   *'I              E X C I T E R S     D A T A                 I',
   */,4X,60('-'))
    DO 10 I     =       1,NG
    GO TO (35,35,3,5,9,7),TYPEX(I)
35 WRITE(IP,2)I,TYPEX(I),TA(I),TA2(I),TA3(I),TE(I),TF(I),GA(I),
   *XKE(I),XKF(I),S1(I),S2(I),EFDMAX(I),EFDMIN(I),VRMAX(I),VRMIN(I)
  2 FORMAT(1H ,3X,
```

```
     *'I                          EXCITER NO. ',I2,24X,'I',/,4X,60('-'),/,4X,
     *'I    TYPE',27X,'=',16X,I2,4X,'I',/,4X,
     *'I    TA',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    TB',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    TC',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    TE',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    TF',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    KA',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    KE',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    KF',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    S1',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    S2',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    EFDMAX',25X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    EFDMIN',25X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    VRMAX',26X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    VRMIN',26X,'= ',7X,F10.4,4X,'I',/,4X,60('-'))
      GO TO 9
    3 CONTINUE
      WRITE(IP,4)I,TYPEX(I),TA(I),TE(I),TF(I),GA(I),
     *XKF(I),GP(I),GI(I),VBMAX(I),EFDMAX(I),EFDMIN(I),VRMAX(I),VRMIN(I)
    4 FORMAT(1H ,3X,60('-'),/,4X,
     *'I                          EXCITER NO. ',I2,24X,'I',/,4X,60('-'),/,4X,
     *'I    TYPE',27X,'=',16X,I2,4X,'I',/,4X,
     *'I    TA',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    TE',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    TF',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    KA',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    KF',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    KP',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    KI',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    VBMAX',26X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    EFDMAX',25X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    EFDMIN',25X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    VRMAX',26X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    VRMIN',26X,'= ',7X,F10.4,4X,'I',/,4X,60('-'))
      GO TO 9
    5 CONTINUE
      WRITE(IP,6)I,TYPEX(I),TE(I),TA(I),XKE(I),S1(I),S2(I),
```

```
     *EFDMAX(I),EFDMIN(I),VRMAX(I),VRMIN(I)
    6 FORMAT(1H ,3X,60('-'),/,4X,
     *'I                    EXCITER NO. ',I2,24X,'I',/,4X,60('-'),/,4X,
     *'I    TYPE',27X,'=',16X,I2,4X,'I',/,4X,
     *'I    TE',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    TRH',28X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    KE',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    S1',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    S2',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    EFDMAX',25X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    EFDMIN',25X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    VRMAX',26X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    VRMIN',26X,'= ',7X,F10.4,4X,'I',/,4X,60('-'))
      GO TO 9
    7 CONTINUE
      WRITE(IP,8)I,TYPEX(I),TA(I),TA2(I),TA3(I),TF(I),GA(I),
     *XKF(I),EFDMAX(I),EFDMIN(I),VRMAX(I),VRMIN(I)
    8 FORMAT(1H ,3X,60('-'),/,4X,
     *'I                    EXCITER NO. ',I2,24X,'I',/,4X,60('-'),/,4X,
     *'I    TYPE',27X,'=',16X,I2,4X,'I',/,4X,
     *'I    TA',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    TB',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    TC',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    TF',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    KA',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    KF',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    EFDMAX',25X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    EFDMIN',25X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    VRMAX',26X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    VRMIN',26X,'= ',7X,F10.4,4X,'I',/,4X,60('-'))
    9 CONTINUE
   10 CONTINUE
   11 CONTINUE
      RETURN
      END
C     SUBROUTINE PSSWRITE
      SUBROUTINE PSSWRITE(NG,IPSS,XG1,XT1,XT3,XT4,XT5,XT6,XT7,XT8,
     *VL1,VL2,IP)
```

```
      DIMENSION XG1(1),XT1(1),XT3(1),XT4(1),XT5(1),XT6(1),XT7(1),
     *XT8(1),VL1(1),VL2(1)
      INTEGER IPSS(1)
      IJ          =       0
      DO 2 I      =       1,NG
      IF(IPSS(I).EQ.0)GO TO 1
      IJ          =       IJ + 1
    1 CONTINUE
    2 CONTINUE
      IF(IJ.EQ.0)GO TO 7
      WRITE(IP,3)
    3 FORMAT(1H ,3X,60('-'),/,4X,
     *'I                POWER SYSTEM STABILIZER                    I',
     */,4X,60('-'))
      DO 6 I      =       1,NG
      IF(IPSS(I).NE.1)GO TO 5
      WRITE(IP,4)I,XG1(I),XT1(I),XT3(I),XT4(I),XT5(I),XT6(I),XT7(I),
     *XT8(I),VL1(I),VL2(I)
    4 FORMAT(1H ,3X,
     *'I                  PSS NO. ',4X,I2,24X,'I',/,4X,60('-'),/,4X,
     *'I    K1',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    T1',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    T3',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    T4',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    T5',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    T6',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    T7',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    T8',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    L1',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    L2',29X,'= ',7X,F10.4,4X,'I',/,4X,60('-'))
    5 CONTINUE
    6 CONTINUE
    7 CONTINUE
      RETURN
      END
C     SUBROUTINE GOVWRITE
      SUBROUTINE GOVWRITE(NG,IGOV,G1,G2,G3,G4,T1,T2,T3,T4,T5,T6,IP)
      INTEGER IGOV(1)
```

```
      DIMENSION G1(1),G2(1),G3(1),G4(1),T1(1),T2(1),T3(1),T4(1),T5(1),
     *T6(1)
      IX            =       0
      DO 2 I        =       1,NG
      IF(IGOV(I).EQ.0)GO TO 1
      IX            =       IX + 1
    1 CONTINUE
    2 CONTINUE
      IF(IX.EQ.0)GO TO 7
      WRITE(IP,3)
    3 FORMAT(1H ,3X,60('-'),/,4X,
     *'I                SPEED-GOVERNING SYSTEM DATA                 I',
     */,4X,60('-'))
      DO 6 I        =       1,NG
      IF(IGOV(I).NE.1)GO TO 5
      WRITE(IP,4)I,G1(I),G2(I),G3(I),G4(I),T1(I),T2(I),T3(I),T4(I),
     *T5(I),T6(I)
    4 FORMAT(1H ,3X,
     *'I                 SPEED GOVERNER NO.  ',I2,20X,'I',
     */,4X,60('-'),/,4X,
     *'I    G1',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    G2',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    G3',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    G4',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    T1',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    T2',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    T3',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    T4',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    T5',29X,'= ',7X,F10.4,4X,'I',/,4X,
     *'I    T6',29X,'= ',7X,F10.4,4X,'I',/,4X,60('-'))
    5 CONTINUE
    6 CONTINUE
    7 CONTINUE
      RETURN
      END
C     SUBROUTINE BUSWRITE
      SUBROUTINE BUSWRITE(ANG,VB,P,PL,Q,QL,TYPE,YCHARG,NB,IP)
      DIMENSION P(1),PL(1),Q(1),QL(1),VB(1),ANG(1)
```

```
      INTEGER TYPE(1)
      COMPLEX YCHARG(1)
      WRITE(IP,1)
    1 FORMAT(1H ,3X,76('-') ,/,4X,'I',29X,'B U S   D A T A',
     *30X,'I',/,4X,76('-'),/,4X,'I',5X,'I',6X,'I',3X,'GENERATION',
     *4X,'I',6X,'LOAD',7X,'I',7X,'I',9X,'I',7X,'I',/,4X,'I',1X,'NO.',
     *1X,'I',1X,'TYPE',1X,'I',17('-'),'+',17('-'),'I',2X,'VOLT',1X,
     *'I',2X,'PHASE',2X,'I',2X,'CAP',2X,'I',/,4X,'I',5X,'I',6X,'I',
     *4X,'P',3X,'I',4X,'Q',3X,'I',4X,'P',3X,'I',4X,'Q',3X,'I',7X,'I',
     *9X,'I',7X,'I',/,4X,76('-'))
      DO 3 I        =        1,NB
      YCH           =        AIMAG(YCHARG(I))
      WRITE(IP,2)I,TYPE(I),P(I),Q(I),PL(I),QL(I),VB(I),ANG(I),YCH
    2 FORMAT(1H ,3X,'I ',I2,'  I  ',I2,'  I ',4(F6.2,' I '),
     *F5.2,' I ',F7.3,' I ',F5.2,' I',/,4X,76('-'))
    3 CONTINUE
      RETURN
      END
C     SUBROUTINE WRITE
      SUBROUTINE WRITE
      COMMON /CX1/ANG(10),BW(10),BX(10),CUI(10),CUR(10),DAMP(10),
     *DEL(10),DELV1(10),DELV2(10),EFDMAX(10),EFDMIN(10),EGD(10),
     *EGQ(10),GA(10),GC(10),GI(10),GP(10),GV(10),GW(10),GX(10),
     *G1(10),G2(10),G3(10),G4(10),HBASE(10),HC(10),P(15),PL(15),
     *PM(15),PHSF(10),Q(15),QL(15),RA(10),RL(10),RT(10),RV(10),
     *S1(10),S2(10),SAT1(10),SAT2(10),TA(10),TA2(10),TA3(10),
     *TAP(10),TDOP(10),TDOPP(10),TE(10),TF(10),TQOP(10),TQOPP(10),
     *TR(10),T1(10),T2(10),T3(10),T4(10),T5(10),T6(10),VB(15),
     *VBMAX(10),VHMAX(10),VHMIN(10),VRMAX(10),VRMIN(10),VL1(10),
     *VL2(10),XD(10),XDP(10),XDPP(10),XKE(10),XKEP(10),XKF(10),
     *XKFP(10),XL(10),XP(10),XQ(10),XQP(10),XQPP(10),XS(10),XT(10),
     *XV(10),XG1(10),XT1(10),XT3(10),XT4(10),XT5(10),XT6(10),XT7(10),
     *XT8(10),YSHUNT(10)
      INTEGER GB(5),TYPEX(5),SB(20),EB(20),ST(10),ET(10),IMAG(20),
     *TYPE(20),LIST(40),NEXT(800),FAR(800),NCONN(40),MODEL(8),
     *IPSS(10),IGOV(10)
      COMPLEX Z(10,10),YCHARG(20),SPD(30),RH(30),SPC(30,4),TC(20),
     *EE(30),E(15),ASELF(40),AMUT(800)
```

```
      COMMON /CX2/GB,TYPEX,SB,EB,ST,ET,IMAG,TYPE,LIST,NEXT,FAR,NCONN,
     *MODEL,IPSS,IGOV
      COMMON /CX3/Z,YCHARG,SPD,RH,SPC,TC,EE,E,ASELF,AMUT
      COMMON /CX4/REXC,XEXC,RGOV,XGOV,RPSS,XPSS,FREQCY,C1DS,C1QS,C2DS,
     *C2QS,CV1,CV2,CV3,CV4,EFDO,SIGMA,OMEGA,TOL,PI,WD,FMIN,FINC,FO,
     *SBASE,Y11,Y12,Y21,Y22,C1D,C2D,C3D,C4D,C5D,C6D,C7D,C1Q,C2Q,C3Q,
     *C4Q,C5Q,C6Q,C7Q,TSDO,TFDO,TSQO,TFQO,QAXISX,DAXISX
      COMMON /CX5/NB,NL,NT,NG,MAX,MAXF,IDRIVE,IC,ISEL
      I1          =       0
      I2          =       0
      I3          =       0
      DO 1 I      =       1,NG
      IF(TYPEX(I).NE.0)I1 = I1 + 1
      IF(IPSS(I).NE.0)I2 = I2 + 1
      IF(IGOV(I).NE.0)I3 = I3 + 1
    1 CONTINUE
    2 FORMAT(8I5)
    3 FORMAT(8F10.4)
    4 FORMAT(2I5,5F10.4)
    5 FORMAT(3F10.4,4I5)
    6 FORMAT(7F10.4,I5)
      WRITE(7,2)NB,NL,NT,NG,MAX,IDRIVE,MAXF
      WRITE(7,3)FREQCY,SBASE,TOL,WD,FO,FMIN,FINC
      DO 8 I      =       1,NL
      WRITE(7,4)SB(I),EB(I),RL(I),XS(I),YSHUNT(I)
    8 CONTINUE
      DO 9 I      =       1,NT
      WRITE(7,4)ST(I),ET(I),RT(I),XT(I),TAP(I),PHSF(I)
    9 CONTINUE
      DO 10 I     =       1,NB
      YCH         =       AIMAG(YCHARG(I))
      WRITE(7,6)P(I),PL(I),Q(I),QL(I),VB(I),ANG(I),YCH,TYPE(I)
   10 CONTINUE
      DO 11 I     =       1,NG
      WRITE(7,3)RA(I),XL(I),XD(I),XDP(I),XDPP(I),XQ(I),XQP(I),XQPP(I)
   11 CONTINUE
      DO 12 I     =       1,NG
      WRITE(7,3)TDOP(I),TDOPP(I),TQOP(I),TQOPP(I),HC(I),HBASE(I),DAMP(I)
```

```
 12 CONTINUE
    DO 13 I       =       1,NG
    WRITE(7,5)XP(I),SAT1(I),SAT2(I),TYPEX(I),IPSS(I),IGOV(I),GB(I)
 13 CONTINUE
    IF(I1.EQ.0)GO TO 18
    DO 14 I       =       1,NG
    IF(TYPEX(I).EQ.0)GO TO 14
    WRITE(7,3)GA(I),TA(I),VRMAX(I),VRMIN(I),RV(I),XV(I),TA2(I),TA3(I)
 14 CONTINUE
    DO 17 I       =       1,NG
    IF(TYPEX(I).EQ.0)GO TO 16
    IF(TYPEX(I).EQ.3)GO TO 15
    WRITE(7,3)XKE(I),TE(I),S1(I),S2(I),XKF(I),TF(I),TR(I)
    GO TO 16
 15 WRITE(7,3)XKE(I),TE(I),GP(I),GI(I),VBMAX(I),XKF(I),TF(I),TR(I)
 16 CONTINUE
 17 CONTINUE
    DO 200 I      =       1,NG
    IF(TYPEX(I).EQ.0)GO TO 199
    WRITE(7,3)EFDMAX(I),EFDMIN(I)
199 CONTINUE
200 CONTINUE
 18 CONTINUE
    IF(I2.EQ.0)GO TO 23
    DO 20 I       =       1,NG
    IF(IPSS(I).NE.1)GO TO 19
    WRITE(7,3)XG1(I),XT1(I),XT3(I),XT4(I),XT5(I),XT6(I),XT7(I),
   *XT8(I)
 19 CONTINUE
 20 CONTINUE
    DO 22 I       =       1,NG
    IF(IPSS(I).NE.1)GO TO 21
    WRITE(7,3)VL1(I),VL2(I)
 21 CONTINUE
 22 CONTINUE
 23 CONTINUE
    IF(I3.EQ.0)GO TO 28
    DO 25 I       =       1,NG
```

```
      IF(IGOV(I).NE.1)GO TO 24
      WRITE(7,3)G1(I),T1(I),T2(I),T3(I)
   24 CONTINUE
   25 CONTINUE
      DO 27 I      =       1,NG
      IF(IGOV(I).NE.1)GO TO 26
      WRITE(7,3)G2(I),T4(I),T5(I),G3(I),G4(I),T6(I)
   26 CONTINUE
   27 CONTINUE
   28 CONTINUE
      RETURN
      END
C     SUBROUTINE GENWRITE
      SUBROUTINE GENWRITE(NG,HBASE,RA,XD,XQ,XDP,XQP,XDPP,
     *XQPP,XL,TDOP,TQOP,TDOPP,TQOPP,HC,DAMP,GB,IP)
      DIMENSION HBASE(1),RA(1),XD(1),XQ(1),XDP(1),XQP(1),
     *XDPP(1),XQPP(1),XL(1),TDOP(1),TQOP(1),TDOPP(1),TQOPP(1),
     *HC(1),DAMP(1)
      INTEGER GB(1)
      ISTR         =       1
      IEND         =       5
      INTER        =       IEND - ISTR
      IF(IEND.GT.NG)IEND=NG
    1 CONTINUE
      WRITE(IP,2)
    2 FORMAT(1H ,3X,69('-'),/,4X,
     *'I                    G E N E R A T O R S    D A T A',
     *'                   I',/,4X,69('-'))
      WRITE(IP,3)(GB(I),I=ISTR,IEND)
    3 FORMAT(1H ,3X,'I  BUS NO.    I',5(4X,I2,4X,'I'),/,4X,69('-'))
      WRITE(IP,4)(HBASE(I),I=ISTR,IEND)
    4 FORMAT(1H ,3X,'I  BASE MVA  I',5(1X,F8.4,1X,'I')/,4X,69('-'))
      WRITE(IP,5)(XD(I),I=ISTR,IEND)
    5 FORMAT(1H ,3X,'I     XD     I',5(1X,F8.4,1X,'I')/,4X,69('-'))
      WRITE(IP,6)(XDP(I),I=ISTR,IEND)
    6 FORMAT(1H ,3X,'I     XDP    I',5(1X,F8.4,1X,'I')/,4X,69('-'))
      WRITE(IP,7)(XDPP(I),I=ISTR,IEND)
    7 FORMAT(1H ,3X,'I     XDPP   I',5(1X,F8.4,1X,'I')/,4X,69('-'))
```

```
      WRITE(IP,8)(XQ(I),I=ISTR,IEND)
    8 FORMAT(1H ,3X,'I      XQ      I',5(1X,F8.4,1X,'I')/,4X,69('-'))
      WRITE(IP,9)(XQP(I),I=ISTR,IEND)
    9 FORMAT(1H ,3X,'I      XQP     I',5(1X,F8.4,1X,'I')/,4X,69('-'))
      WRITE(IP,10)(XQPP(I),I=ISTR,IEND)
   10 FORMAT(1H ,3X,'I      XQPP    I',5(1X,F8.4,1X,'I')/,4X,69('-'))
      WRITE(IP,11)(XL(I),I=ISTR,IEND)
   11 FORMAT(1H ,3X,'I      XL      I',5(1X,F8.4,1X,'I')/,4X,69('-'))
      WRITE(IP,12)(RA(I),I=ISTR,IEND)
   12 FORMAT(1H ,3X,'I      RG      I',5(1X,F8.4,1X,'I')/,4X,69('-'))
      WRITE(IP,13)(TDOP(I),I=ISTR,IEND)
   13 FORMAT(1H ,3X,'I     TDOP     I',5(1X,F8.4,1X,'I')/,4X,69('-'))
      WRITE(IP,14)(TDOPP(I),I=ISTR,IEND)
   14 FORMAT(1H ,3X,'I     TDOPP    I',5(1X,F8.4,1X,'I')/,4X,69('-'))
      WRITE(IP,15)(TQOP(I),I=ISTR,IEND)
   15 FORMAT(1H ,3X,'I     TQOP     I',5(1X,F8.4,1X,'I')/,4X,69('-'))
      WRITE(IP,16)(TQOPP(I),I=ISTR,IEND)
   16 FORMAT(1H ,3X,'I     TQOPP    I',5(1X,F8.4,1X,'I')/,4X,69('-'))
      WRITE(IP,17)(HC(I),I=ISTR,IEND)
   17 FORMAT(1H ,3X,'I       H      I',5(1X,F8.4,1X,'I')/,4X,69('-'))
      WRITE(IP,50)(DAMP(I),I=ISTR,IEND)
   50 FORMAT(1H ,3X,'I     DAMP     I',5(1X,F8.4,1X,'I')/,4X,69('-'))
      ISTR          =        IEND + 1
      IEND          =        ISTR + INTER
      IF(ISTR.GT.NG)GO TO 44
      IF(IEND.GT.NG)IEND=NG
      GO TO 1
   44 CONTINUE
      RETURN
      END
C     SUBROUTINE GMODEL
      SUBROUTINE GMODEL(NG,SBASE,HBASE,RA,XD,XQ,XDP,XQP,XDPP,
     *XQPP,XL,TDOP,TQOP,TDOPP,TQOPP,HC,DAMP,MODEL)
      DIMENSION HBASE(1),RA(1),XD(1),XQ(1),XDP(1),XQP(1),
     *XDPP(1),XQPP(1),XL(1),TDOP(1),TQOP(1),TDOPP(1),TQOPP(1),
     *HC(1),DAMP(1),MODEL(1)
      T             =        1.0E-15
      DO 1 I        =        1,NG
```

```
      RATIO       =       SBASE/HBASE(I)
      RA(I)       =       RA(I)*RATIO
      XD(I)       =       XD(I)*RATIO
      XQ(I)       =       XQ(I)*RATIO
      XDP(I)      =       XDP(I)*RATIO
      XQP(I)      =       XQP(I)*RATIO
      XDPP(I)     =       XDPP(I)*RATIO
      XQPP(I)     =       XQPP(I)*RATIO
      XL(I)       =       XL(I)*RATIO
      HC(I)       =       HC(I)/RATIO
      DAMP(I)     =       DAMP(I)/RATIO
      MODEL(I)    =       0
      IF(TDOP(I).LE.T.AND.TDOPP(I).LE.T.AND.TQOP(I).LE.T.
     *AND.TQOPP(I).LE.T)MODEL(I)=1
      IF(TDOP(I).GT.T.AND.TDOPP(I).LE.T.AND.TQOP(I).LE.T.
     *AND.TQOPP(I).LE.T)MODEL(I)=2
      IF(TDOP(I).GT.T.AND.TDOPP(I).GT.T.AND.TQOP(I).LE.T.
     *AND.TQOPP(I).GT.T)MODEL(I)=3
      IF(TDOP(I).GT.T.AND.TDOPP(I).LE.T.AND.TQOP(I).GT.T.
     *AND.TQOPP(I).LE.T)MODEL(I)=4
      IF(TDOP(I).GT.T.AND.TDOPP(I).GT.T.AND.TQOP(I).GT.T.
     *AND.TQOPP(I).GT.T)MODEL(I)=5
    1 CONTINUE
      RETURN
      END
C     SUBROUTINE INTIAL
      SUBROUTINE INTIAL(NG,RA,XD,XQ,XDP,XQP,XDPP,XQPP,XL,
     *MODEL,P,Q,PM,GB,DEL,CUR,CUI,EGD,EGQ,SAT1,SAT2,XP,E)
      DIMENSION RA(1),XD(1),XQ(1),XDP(1),XQP(1),
     *XDPP(1),XQPP(1),XL(1),
     *P(1),Q(1),PM(1),DEL(1),
     *CUR(1),CUI(1),EGD(1),EGQ(1),SAT1(1),SAT2(1),XP(1)
      COMPLEX E(1)
      INTEGER GB(1),MODEL(1)
      DO 4 I      =       1,NG
      J           =       GB(I)
      ER          =       REAL(E(J))
      EI          =       AIMAG(E(J))
```

```
      VTS          =       ER*ER + EI*EI
      VT           =       SQRT(VTS)
      AMAG         =       0.0
      IF(VT.GT.0.0)AMAG =  ATAN2(EI,ER)
      CUR(I)       =       (ER*P(J) + EI*Q(J))/VTS
      CUI(I)       =       (EI*P(J) - ER*Q(J))/VTS
      PM(I)        =       P(J) + (CUR(I)**2 + CUI(I)**2)*RA(I)
      IF(MODEL(I).EQ.1)GO TO  2
      SQZ          =       0.0
      BSAT         =       0.0
      ASAT         =       SAT1(I)
      IF(SAT1(I).EQ.0.0)GO TO 1
      BSAT         =       ALOG(SAT2(I)/SAT1(I))/ALOG(1.2)
      IF(XP(I).EQ.0.0)XP(I) = XDP(I)
      CMAGS        =       (P(J)**2 + Q(J)**2)/VTS
      FLUXM        =       VTS + CMAGS*(RA(I)**2 + XP(I)**2) +
     *2.0*RA(I)*P(J) + 2.0*XP(I)*Q(J)
      FLUXM        =       SQRT(FLUXM)
      SQZ          =       ASAT*FLUXM**BSAT
    1 XQS          =       XQ(I)
      C3Q          =       0.0
      C5Q          =       0.0
      IF(MODEL(I).EQ.4)C3Q = XQ(I) - XQP(I)
      IF(MODEL(I).EQ.3)C5Q = XQ(I) - XQPP(I)
      IF(MODEL(I).EQ.5)C3Q = (XQ(I) - XQP(I))*(XQPP(I) - XL(I))/
     *(XQP(I) - XL(I))
      IF(MODEL(I).EQ.5)C5Q = (XQ(I)-XL(I))*(XQP(I)-XQPP(I))/
     *(XQP(I)-XL(I))
      IF(MODEL(I).EQ.4)XQS =  (XQ(I) + SQZ*XQP(I))/(1.0 + SQZ)
      IF(MODEL(I).EQ.5)XQS =  XQPP(I) + C5Q + C3Q/(1.0 + SQZ)
      DNUM         =       EI + RA(I)*CUI(I) + XQS*CUR(I)
      DDEN         =       ER + RA(I)*CUR(I) - XQS*CUI(I)
      DEL(I)       =       0.0
      IF(DNUM.GT.0.0)DEL(I) = ATAN2(DNUM,DDEN)
      SND          =       SIN(DEL(I))
      CSD          =       COS(DEL(I))
      ED           =       ER*SND - EI*CSD
      EQ           =       ER*CSD + EI*SND
```

```
      CUD           =       CUR(I)*SND - CUI(I)*CSD
      CUQ           =       CUR(I)*CSD + CUI(I)*SND
      IF(MODEL(I).EQ.3.OR.MODEL(I).EQ.5)DAXIS=XDPP(I)
      IF(MODEL(I).EQ.3.OR.MODEL(I).EQ.5)QAXIS=XQPP(I)
      IF(MODEL(I).EQ.2.OR.MODEL(I).EQ.4)DAXIS=XDP(I)
      IF(MODEL(I).EQ.2)QAXIS = XQ(I)
      IF(MODEL(I).EQ.4)QAXIS = XQP(I)
      EGD(I)        =       ED + RA(I)*CUD - QAXIS*CUQ
      EGQ(I)        =       EQ + RA(I)*CUQ + DAXIS*CUD
      GO TO 3
    2 CONTINUE
C     MODEL 1 (CLASSICAL MODEL)
      EGR           =       RA(I)*CUR(I) - XDP(I)*CUI(I) + ER
      EGI           =       RA(I)*CUI(I) + XDP(I)*CUR(I) + EI
      DEL(I)        =       0.0
      IF(EGI.GT.0.0)DEL(I)=ATAN2(EGI,EGR)
      EGD(I)        =       0.0
      EGQ(I)        =       SQRT(EGR*EGR + EGI*EGI)
    3 CONTINUE
    4 CONTINUE
      RETURN
      END
C     SUBROUTINE CONSTA
      SUBROUTINE CONSTA(K,RA,XD,XQ,XDP,XQP,XDPP,XQPP,XL,TDOP,TQOP,
     *TDOPP,TQOPP,MODEL)
      DIMENSION RA(1),XD(1),XQ(1),XDP(1),XQP(1),XDPP(1),XQPP(1),XL(1),
     *TDOP(1),TQOP(1),TDOPP(1),TQOPP(1),MODEL(1)
      COMMON /CX4/REXC,XEXC,RGOV,XGOV,RPSS,XPSS,FREQCY,C1DS,C1QS,C2DS,
     *C2QS,CV1,CV2,CV3,CV4,EFDO,SIGMA,OMEGA,TOL,PI,WD,FMIN,FINC,FO,
     *SBASE,Y11,Y12,Y21,Y22,C1D,C2D,C3D,C4D,C5D,C6D,C7D,C1Q,C2Q,C3Q,
     *C4Q,C5Q,C6Q,C7Q,TSDO,TFDO,TSQO,TFQO,QAXISX,DAXISX
      D1            =       XD(K) - XDP(K)
      D2            =       XDP(K) - XDPP(K)
      D3            =       XDP(K) - XL(K)
      D4            =       XDPP(K) - XL(K)
      D5            =       XD(K) - XL(K)
      Q1            =       XQ(K) - XQP(K)
      Q2            =       XQP(K) - XQPP(K)
```

```
      Q3              =       XQP(K) - XL(K)
      Q4              =       XQPP(K) - XL(K)
      Q5              =       XQ(K) - XL(K)
      Q6              =       XQ(K) - XQPP(K)
      A               =       1.0E-9
      IF(XL(K).GT.A.AND.D1.GT.A.AND.D3.GT.A.AND.Q5.GT.A)GO TO 2
      WRITE(*,1)K
    1 FORMAT(1H ,5X,'GENERATOR NO. ',I2,'  DOES NOT HAVE',/,6X,
     *'PROPER REACTANCE')
      STOP
    2 CONTINUE
      IF(MODEL(K).EQ.2)GO TO 3
      IF(MODEL(K).EQ.4.AND.Q1.GT.A.AND.Q3.GT.A)GO TO 3
      IF(MODEL(K).EQ.3.AND.D2.GT.A.AND.D4.GT.A.AND.Q4.GT.A.AND.
     *Q6.GT.A)GO TO 3
      IF(MODEL(K).EQ.5.AND.D2.GT.A.AND.D4.GT.A.AND.Q4.GT.A.AND.
     *Q1.GT.A.AND.Q2.GT.A)GO TO 3
      WRITE(*,1)K
      STOP
    3 CONTINUE
C     FOR MODEL 2,3, AND 4
      C1Q             =       1.0
      C2Q             =       0.0
      C3Q             =       0.0
      C4Q             =       0.0
      C5Q             =       0.0
      C6Q             =       0.0
      C7Q             =       0.0
      TFQO            =       0.0
      TSQO            =       0.0
      GO TO(10,5,4,5,4),MODEL(K)
C     FOR MODEL 3 AND 5
    4 CONTINUE
      C1D             =       1.0 + D1*D2/D3**2
      C2D             =       (D1/D3)*(D2/D3 + D4/D5)
      C3D             =       D1*D4/D3
      C4D             =       D5*D2/D3**2
      C5D             =       D5*D2/D3
```

```
      C6D          =      D4/D3
      C7D          =      D2/D3 + D4/D5
      DAXISX       =      XDPP(K)
      QAXISX       =      XQPP(K)
      RAD          =      (TDOP(K) + TDOPP(K))**2 -
     *                    4.0*C1D*TDOP(K)*TDOPP(K)
      RAD          =      SQRT(RAD)
      TFDO         =      0.5*(TDOP(K) + TDOPP(K)+ RAD)
      TSDO         =      0.5*(TDOP(K) + TDOPP(K)- RAD)
      GO TO 100
    5 CONTINUE
C     MODEL 2 AND 4
      C1D          =      1.0
      C2D          =      0.0
      C3D          =      XD(K) - XDP(K)
      C4D          =      0.0
      C5D          =      0.0
      C6D          =      1.0
      C7D          =      0.0
      TFDO         =      TDOP(K)
      TSDO         =      0.0
      DAXISX       =      XDP(K)
  100 GO TO(10,7,8,9,6),MODEL(K)
C     MODEL 5
    6 CONTINUE
      C1Q          =      1.0 + Q1*Q2/Q3**2
      C2Q          =      (Q1/Q3)*(Q2/Q3 + Q4/Q5)
      C3Q          =      Q1*Q4/Q3
      C4Q          =      Q5*Q2/Q3**2
      C5Q          =      Q5*Q2/Q3
      C6Q          =      Q4/Q3
      C7Q          =      Q2/Q3 + Q4/Q5
      RAD          =      (TQOP(K) + TQOPP(K))**2 -
     *                    4.0*C1Q*TQOP(K)*TQOPP(K)
      RAD          =      SQRT(RAD)
      TFQO         =      0.5*(TQOP(K) + TQOPP(K) + RAD)
      TSQO         =      0.5*(TQOP(K) + TQOPP(K) - RAD)
      GO TO 10
```

```
C     MODEL 2
    7 CONTINUE
      QAXISX        =       XQ(K)
      GO TO 10
C     MODEL 3
    8 CONTINUE
      C5Q           =       XQ(K) - XQPP(K)
      C7Q           =       1.0
      TSQO          =       TQOPP(K)
      GO TO 10
C     MODEL 4
    9 CONTINUE
      C3Q           =       XQ(K) - XQP(K)
      C6Q           =       1.0
      TFQO          =       TQOP(K)
      QAXISX        =       XQP(K)
   10 CONTINUE
      RETURN
      END
C     SUBROUTINE ACONST
      SUBROUTINE ACONST(K,RA,XD,XQ,XDP,XQP,XDPP,XQPP,XL,
     *MODEL,P,Q,PM,GB,DEL,CUR,CUI,EGD,EGQ,SAT1,SAT2,XP,E,
     *TYPEX,RV,XV,Z)
      DIMENSION RA(1),XD(1),XQ(1),XDP(1),XQP(1),
     *XDPP(1),XQPP(1),XL(1),
     *P(1),Q(1),PM(1),DEL(1),
     *CUR(1),CUI(1),EGD(1),EGQ(1),SAT1(1),SAT2(1),XP(1),
     *RV(1),XV(1)
      COMPLEX E(1),Z(10,10)
      INTEGER TYPEX(1),GB(1),MODEL(1)
      COMMON /CX4/REXC,XEXC,RGOV,XGOV,RPSS,XPSS,FREQCY,C1DS,C1QS,C2DS,
     *C2QS,CV1,CV2,CV3,CV4,EFDO,SIGMA,OMEGA,TOL,PI,WD,FMIN,FINC,FO,
     *SBASE,Y11,Y12,Y21,Y22,C1D,C2D,C3D,C4D,C5D,C6D,C7D,C1Q,C2Q,C3Q,
     *C4Q,C5Q,C6Q,C7Q,TSDO,TFDO,TSQO,TFQO,QAXISX,DAXISX
      DO 9 L1       =       3,10
      DO 8 M1       =       1,10
      Z(L1,M1)      =       CMPLX(0.0,0.0)
    8 CONTINUE
```

```
      Z(L1,L1)      =       CMPLX(1.0,0.0)
    9 CONTINUE
      PI            =       3.141592653
      OMEGAB        =       2.0*PI*FREQCY
      J             =       GB(K)
      ER            =       REAL(E(J))
      EI            =       AIMAG(E(J))
      SND           =       SIN(DEL(K))
      CSD           =       COS(DEL(K))
      IF(MODEL(K).EQ.1)GO TO 1
      CUD           =       CUR(K)*SND - CUI(K)*CSD
      CUQ           =       CUR(K)*CSD + CUI(K)*SND
      BSAT          =       0.0
      IF(ABS(SAT2(K)).LE.0.0)GO TO 77
      IF(SAT1(K).GT.0.0)BSAT = ALOG(SAT2(K)/SAT1(K))/ALOG(1.2)
   77 CONTINUE
      SQZ           =       SAT1(K)*ABS(EGQ(K))**BSAT
      SN2D          =       SIN(2.0*DEL(K))
      CS2D          =       COS(2.0*DEL(K))
      TEMP1         =       CUR(K)*CS2D + CUI(K)*SN2D
      TEMP2         =       CUR(K)*SN2D - CUI(K)*CS2D
      R11           =       RA(K) - (QAXISX -DAXISX)*SND*CSD
      X12           =       -QAXISX*SND**2 - DAXISX*CSD**2
      X21           =       QAXISX*CSD**2 + DAXISX*SND**2
      R22           =       RA(K) + (QAXISX - DAXISX)*SND*CSD
      DET           =       R11*R22 -X12*X21
      Y11           =       R22/DET
      Y22           =       R11/DET
      Y12           =       -X12/DET
      Y21           =       -X21/DET
      AR            =       EGD(K)*CSD - EGQ(K)*SND +
     *                      (QAXISX -DAXISX)*TEMP1
      AI            =       EGD(K)*SND + EGQ(K)*CSD +
     *                      (QAXISX - DAXISX)*TEMP2
      IF(MODEL(K).EQ.2.OR.MODEL(K).EQ.4)
     *EFDO          =       (1.0 + SQZ)*EGQ(K) + (XD(K) -XDP(K))*CUD
      IF(MODEL(K).EQ.3.OR.MODEL(K).EQ.5)
     *EFDO          =       (1.0 +SQZ)*(EGQ(K) + C5D*CUD) + C3D*CUD
```

```
TEMP1       =     1.0 + (1.0 +BSAT)*SQZ
C1DS        =     C1D*TEMP1
C1QS        =     C1Q*TEMP1
C2DS        =     C2D*TEMP1
C2QS        =     C2Q*TEMP1
EGR         =     EGD(K)*SND + EGQ(K)*CSD
EGI         =     -EGD(K)*CSD + EGQ(K)*SND
CALL CVV(K,TYPEX,GB,CUR,CUI,RV,XV,E)
Z(6,3)      =     CMPLX(CV3*Y11 + CV4*Y21,0.0)
Z(6,4)      =     CMPLX(CV3*Y12 + CV4*Y22,0.0)
Z(6,1)      =     CMPLX(CV1,0.0)- Z(6,3)
Z(6,2)      =     CMPLX(CV2,0.0) - Z(6,4)
TEMP1       =     ER + 2.0*CUR(K)*RA(K)
TEMP2       =     EI + 2.0*CUI(K)*RA(K)
Z(5,1)      =     CMPLX(CUR(K) - TEMP1*Y11 - TEMP2*Y21,0.0)
Z(5,2)      =     CMPLX(CUI(K) - TEMP1*Y12 - TEMP2*Y22,0.0)
Z(5,3)      =     CMPLX(TEMP1*Y11 + TEMP2*Y21,0.0)
Z(5,4)      =     CMPLX(TEMP1*Y12 + TEMP2*Y22,0.0)
TEMP1       =     Y11*SND - Y21*CSD
TEMP2       =     Y22*CSD - Y12*SND
Z(7,1)      =     CMPLX(-C3D*TEMP1,0.0)
Z(7,2)      =     CMPLX(C3D*TEMP2,0.0)
Z(8,1)      =     CMPLX(-C5D*TEMP1,0.0)
Z(8,2)      =     CMPLX(C5D*TEMP2,0.0)
Z(7,3)      =     CMPLX(C3D*TEMP1,0.0)
Z(7,4)      =     CMPLX(-C3D*TEMP2,0.0)
Z(8,3)      =     CMPLX(C5D*TEMP1,0.0)
Z(8,4)      =     CMPLX(-C5D*TEMP2,0.0)
TEMP1       =     Y11*CSD + Y21*SND
TEMP2       =     Y22*SND + Y12*CSD
Z(9,1)      =     CMPLX(-C3Q*TEMP1,0.0)
Z(9,2)      =     CMPLX(-C3Q*TEMP2,0.0)
Z(10,1)     =     CMPLX(-C5Q*TEMP1,0.0)
Z(10,2)     =     CMPLX(-C5Q*TEMP2,0.0)
Z(9,3)      =     CMPLX(C3Q*TEMP1,0.0)
Z(9,4)      =     CMPLX(C3Q*TEMP2,0.0)
Z(10,3)     =     CMPLX(C5Q*TEMP1,0.0)
Z(10,4)     =     CMPLX(C5Q*TEMP2,0.0)
```

```
      Z(3,5)      =     CMPLX(AR*OMEGAB,0.0)
      Z(4,5)      =     CMPLX(AI*OMEGAB,0.0)
      Z(7,5)      =     CMPLX(-C3D*CUQ*OMEGAB,0.0)
      Z(8,5)      =     CMPLX(-C5D*CUQ*OMEGAB,0.0)
      Z(9,5)      =     CMPLX(C3Q*CUD*OMEGAB,0.0)
      Z(10,5)     =     CMPLX(C5Q*CUD*OMEGAB,0.0)
      Z(3,7)      =     CMPLX(-C6D*CSD,0.0)
      Z(3,8)      =     CMPLX(-C7D*CSD,0.0)
      Z(3,9)      =     CMPLX(C6Q*SND,0.0)
      Z(3,10)     =     CMPLX(C7Q*SND,0.0)
      Z(4,7)      =     CMPLX(-C6D*SND,0.0)
      Z(4,8)      =     CMPLX(-C7D*SND,0.0)
      Z(4,9)      =     CMPLX(-C6Q*CSD,0.0)
      Z(4,10)     =     CMPLX(-C7Q*CSD,0.0)
      GO TO 2
    1 CONTINUE
C     MODEL 1
      EGR         =     EGD(K)*SND + EGQ(K)*CSD
      EGI         =     -EGD(K)*CSD + EGQ(K)*SND
      DET         =     RA(K)**2 + XDP(K)**2
      Y11         =     RA(K)/DET
      Y12         =     XDP(K)/DET
      Y21         =     Y12
      Y22         =     Y11
      TEMP1       =     ER + 2.0*CUR(K)*RA(K)
      TEMP2       =     EI + 2.0*CUI(K)*RA(K)
      Z(3,5)      =     CMPLX(-OMEGAB*EGI,0.0)
      Z(4,5)      =     CMPLX(OMEGAB*EGR,0.0)
      Z(5,1)      =     CMPLX(CUR(K) - TEMP1*Y11 - TEMP2*Y21,0.0)
      Z(5,2)      =     CMPLX(CUI(K) - TEMP1*Y12 - TEMP2*Y22,0.0)
      Z(5,3)      =     CMPLX(TEMP1*Y11 + TEMP2*Y21,0.0)
      Z(5,4)      =     CMPLX(TEMP1*Y12 + TEMP2*Y22,0.0)
      TFDO        =     0.0
      TSDO        =     0.0
      TFQO        =     0.0
      TSQO        =     0.0
    2 CONTINUE
      RETURN
```

```
      END
C     SUBROUTINE CVV
      SUBROUTINE CVV(K,TYPEX,GB,CUR,CUI,RV,XV,E)
      DIMENSION CUR(1),CUI(1),RV(1),XV(1)
      INTEGER TYPEX(1),GB(1)
      COMPLEX E(1)
      COMMON /CX4/REXC,XEXC,RGOV,XGOV,RPSS,XPSS,FREQCY,C1DS,C1QS,C2DS,
     *C2QS,CV1,CV2,CV3,CV4,EFDO,SIGMA,OMEGA,TOL,PI,WD,FMIN,FINC,FO,
     *SBASE,Y11,Y12,Y21,Y22,C1D,C2D,C3D,C4D,C5D,C6D,C7D,C1Q,C2Q,C3Q,
     *C4Q,C5Q,C6Q,C7Q,TSDO,TFDO,TSQO,TFQO,QAXISX,DAXISX
C     RV(K)          =       COMPENSATING RESISTANCE(RC)
C     XV(K)          =       COMPENSATING REACTANCE(XC)
      IF(TYPEX(K).LT.1)GO TO 1
      J              =       GB(K)
      ER             =       REAL(E(J))
      EI             =       AIMAG(E(J))
      ECRO           =       ER + RV(K)*CUR(K) - XV(K)*CUI(K)
      ECIO           =       EI + RV(K)*CUI(K) + XV(K)*CUR(K)
      ECO            =       ECRO**2 + ECIO**2
      ECO            =       SQRT(ECO)
      CV1            =       ECRO/ECO
      CV2            =       ECIO/ECO
      CV3            =       RV(K)*CV1 + XV(K)*CV2
      CV4            =       RV(K)*CV2 - XV(K)*CV1
      GO TO 2
    1 CV1            =       0.0
      CV2            =       0.0
      CV3            =       0.0
      CV4            =       0.0
    2 RETURN
      END
C     SUBROUTINE EXCITER
      SUBROUTINE EXCITER(K,TYPEX,GB,GA,TA,VRMIN,VRMAX,RV,XV,TA2,TA3,
     *XKE,TE,S1,S2,EFDMAX,EFDMIN,XKF,TF,TR,XKEP,XKFP,DELV1,DELV2,Q,GP,
     *GI,GC,GV,CURO,CUIO,VBMAX)
      INTEGER TYPEX(1),GB(1)
      DIMENSION GA(1),TA(1),VRMIN(1),VRMAX(1),RV(1),XV(1),TA2(1),TA3(1),
     *XKE(1),TE(1),S1(1),S2(1),EFDMAX(1),EFDMIN(1),XKF(1),TF(1),TR(1),
```

```
     *XKEP(1),XKFP(1),DELV1(1),DELV2(1),Q(1),GP(1),GI(1),GC(1),
     *GV(1),CURO(1),CUIO(1),VBMAX(1)
      COMMON /CX4/REXC,XEXC,RGOV,XGOV,RPSS,XPSS,FREQCY,C1DS,C1QS,C2DS,
     *C2QS,CV1,CV2,CV3,CV4,EFDO,SIGMA,OMEGA,TOL,PI,WD,FMIN,FINC,FO,
     *SBASE,Y11,Y12,Y21,Y22,C1D,C2D,C3D,C4D,C5D,C6D,C7D,C1Q,C2Q,C3Q,
     *C4Q,C5Q,C6Q,C7Q,TSDO,TFDO,TSQO,TFQO,QAXISX,DAXISX
      COMPLEX E(1)
      IF(TYPEX(K).LT.1)GO TO 300
      IF(TYPEX(K).GT.6)GO TO 103
      GO TO (104,104,105,104,103,106),TYPEX(K)
  104 CONTINUE
      A             =       0.0
      IF(S1(K).EQ.A.AND.S2(K).EQ.A.AND.XKE(K).GT.A)GO TO 107
      IF(S2(K).GT.S1(K).AND.S1(K).GT.A.AND.(XKE(K)+S2(K)).GT.A)GO TO 108
  110 WRITE(*,109)K
  109 FORMAT(1H ,2X,'VALUE OF SATURATION FUNCTION AND KE',/,3X,
     *'OF EXCITER AT GEN. NO. ',I2,'IS INCONSISTENT')
      STOP
  107 CONTINUE
C     NO SATURATION
      AEXC          =       0.0
      BEXC          =       0.0
      SEO           =       0.0
      IF(VRMAX(K).GT.A)GO TO 115
      IF(EFDMAX(K).GT.A)GO TO 113
      WRITE(*,114)K
  114 FORMAT(1H ,2X,'AT GEN. NO. ',I2,' VRMAX OR EFDMAX',/,3X,
     *' OF EXCITATION SYSTEM MUST EXCEED ZERO')
      STOP
  113 VRMAX(K)      =       XKE(K)*EFDMAX(K)
      VRMIN(K)      =       XKE(K)*EFDMIN(K)
      GO TO 112
  108 CONTINUE
C     TAKE SATURATION EFFECT INTO ACCOUNT
      BEXC          =       ALOG(S2(K)/S1(K))/ALOG(1.0/0.75)
      IF(ABS(XKE(K)).NE.A.AND.VRMAX(K).GT.A)GO TO 115
      IF(EFDMAX(K).GT.A)GO TO 116
      WRITE(*,117)K
```

```
RMAT(1H ,'AT BUS ',I2,' INVALID KE IS DETECTED')
OP
DMAX(K)     =       VRMAX(K)/(XKE(K)+S2(K))
XC          =       S2(K)/(EFDMAX(K)**BEXC)
)           =       AEXC*(EFDO)**BEXC
 TO 112
XC          =       S2(K)/(EFDMAX(K)**BEXC)
)           =       AEXC*ABS(EFDO)**BEXC
 ABS(XKE(K)).EQ.A)XKE(K)=-SEO
AX(K)       =       (XKE(K)+S2(K))*EFDMAX(K)
N           =       0.0
 ABS(EFDMIN(K)).NE.A)SMIN=AEXC*ABS(EFDMIN(K))**BEXC
IN(K)           =       (XKE(K)+SMIN)*EFDMIN(K)
XKE(K).GT.A)GO TO 112
T           =       -XKE(K)/(1.0 + BEXC)
T           =       ALOG(SCRT/AEXC)/BEXC
T           =       XKE(K)*(BEXC/(1.0 + BEXC))*2.71828**EXPT
VRMIN(K).GT.VCRT)VRMIN(K)=VCRT
TINUE
            =       (XKE(K) + SEO)*EFDO
P(K)        =       XKE(K) + SEO + BEXC*SEO
P(K)        =       XKF(K)
TYPEX(K).EQ.2)XKFP(K)=XKF(K)*XKEP(K)
VRMIN(K).LE.VRO.AND.VRMAX(K).GE.VRO)GO TO 119
TE(*,130)K
MAT(1H ,'GEN. NO. ',I2,' VRO OUT BOUNDS')
P
EFDMIN(K).LE.EFDO.AND.EFDMAX(K).GE.EFDO)GO TO 120
TE(*,131)K
MAT(1H ,'GEN. NO. ',I2,'EFDO OUT OF BOUND')
P
TYPEX(K).NE.4)RETURN
V(K).GT.A)GO TO 118
TE(*,132)K
MAT(1H ,'GEN. NO. ',I2,' KV IS ZERO')
P
BS(DELV2(K)).EQ.A)DELV2(K) = GV(K)
VHMAX(K).GT.A)GO TO 140
```

```
C       VHMIN(K)      =       VRMIN(K)
C       VHMAX(K)      =       VRMAX(K)
  140 CONTINUE
C       VRMAX(K)      =       VHMAX(K)
C       VRMIN(K)      =       VHMIN(K)
        TA2(K)        =       DELV1(K)
        VRO           =       DELV2(K)
        XKEP(K)       =       XKE(K)
        XKFP(K)       =       XKF(K)
        RETURN
  105 TEMP1           =       (GP(K)**2)*(ER**2 +  EI**2)
        TEMP2         =       (GI(K)**2)*(CURO(K)**2 + CUIO(K)**2)
        L             =       GB(K)
        TEMP3         =       2.0*GP(K)*GI(K)*Q(L)
        TEMP4         =       (GC(K)*EFDO)**2
        VEXO          =       TEMP1 + TEMP2 + TEMP3 - TEMP4
        XKEP(K)       =       XKE(K)
        XKFP(K)       =       XKF(K)
        IF(VEXO.GT.A)GO TO 121
        WRITE(*,133)K
  133 FORMAT(1H ,'GEN. NO. ',I2,' INITIAL VEX IS ZERO')
        STOP
  121 VEXO            =       SQRT(VEXO)
        TEMP1         =       (GP(K)*ER - GI(K)*CUIO(K))/VEXO
        TEMP2         =       (GP(K)*EI + GI(K)*CURO(K))/VEXO
        C31           =       GP(K)*TEMP1
        C32           =       GP(K)*TEMP2
        C33           =       GI(K)*TEMP2
        C34           =       GI(K)*TEMP1
        C35           =       (GC(K)**2)*EFDO/VEXO
        IF(EFDO.GT.A.AND.EFDO.EQ.VBMAX(K))GO TO 122
        WRITE(*,131)K
        STOP
  122 VBO             =       XKE(K)*EFDO
        VRO           =       VBO - VEXO
        IF(VRO.GE.VRMIN(K).AND.VRO.LE.VRMAX(K))RETURN
        WRITE(*,130)K
        STOP
```

```
106 CONTINUE
    XKEP(K)       =       XKE(K)
    XKFP(K)       =       XKF(K)
    IF(ABS(EFDMAX(K)).GT.A)VRMAX(K)=EFDMAX(K)
    IF(ABS(EFDMAX(K)).GT.A)VRMAX(K)=EFDMAX(K)
    EFDMAX(K)     =       VRMAX(K)
    EFDMIN(K)     =       VRMIN(K)
    VRO           =       EFDO
    IF(EFDO.GE.EFDMIN(K).AND.EFDO.LE.EFDMAX(K))RETURN
    WRITE(*,131)K
    STOP
103 WRITE(*,134)K
134 FORMAT(1H ,'GEN. NO. ',I2,'VR TYPE NOT VALID')
    STOP
300 CV2           =       0.0
    CV2           =       0.0
    CV3           =       0.0
    CV4           =       0.0
    RETURN
    END
C       SUBROUTINE ZCONST
    SUBROUTINE ZCONST(K,Z,TR,DAMP,HC)
    COMPLEX Z(10,10)
    DIMENSION TR(1),DAMP(1),HC(1)
    COMMON /CX4/REXC,XEXC,RGOV,XGOV,RPSS,XPSS,FREQCY,C1DS,C1QS,C2DS,
   *C2QS,CV1,CV2,CV3,CV4,EFDO,SIGMA,OMEGA,TOL,PI,WD,FMIN,FINC,FO,
   *SBASE,Y11,Y12,Y21,Y22,C1D,C2D,C3D,C4D,C5D,C6D,C7D,C1Q,C2Q,C3Q,
   *C4Q,C5Q,C6Q,C7Q,TSDO,TFDO,TSQO,TFQO,QAXISX,DAXISX
    HORI          =       SIGMA/(SIGMA**2 + OMEGA**2)
    VERTI         =       OMEGA/(SIGMA**2 + OMEGA**2)
    Z(3,5)        =       CMPLX(HORI,VERTI)*Z(3,5)
    Z(4,5)        =       Z(4,5)*CMPLX(HORI,VERTI)
    X             =       DAMP(K) + RGOV -2.0*HC(K)*SIGMA
    T             =       XGOV + 2.0*HC(K)*OMEGA
    Z(5,5)        =       CMPLX(X,T)
    X             =       1.0 - SIGMA*TR(K)
    T             =       OMEGA*TR(K)
    Z(6,5)        =       -CMPLX(X,T)*CMPLX(RPSS,XPSS)
```

```
      Z(6,6)        =       CMPLX((1.0-SIGMA*TR(K)),OMEGA*TR(K))
      Z(7,5)        =       Z(7,5)*CMPLX(HORI,VERTI)
      Z(7,6)        =       CMPLX(-REXC,-XEXC)
      Z(7,7)        =       CMPLX(C1DS,0.0)-CMPLX((SIGMA*TFDO),0.0)+
     *                      CMPLX(0.0,(OMEGA*TFDO))
      Z(7,8)        =       CMPLX(-C2DS,0.0)
      Z(8,5)        =       Z(8,5)*CMPLX(HORI,VERTI)
      Z(8,7)        =       CMPLX(-C4D,0.0)
      Z(8,8)        =       CMPLX(C1D,0.0)-CMPLX(SIGMA*TSDO,0.0)+
     *                      CMPLX(0.0,OMEGA*TSDO)
      Z(9,5)        =       CMPLX(HORI,VERTI)*Z(9,5)
      Z(9,9)        =       CMPLX(C1QS,0.0)-CMPLX(SIGMA*TFQO,0.0)+
     *                      CMPLX(0.0,OMEGA*TFQO)
      Z(9,10)       =       CMPLX(-C2QS,0.0)
      Z(10,5)       =       CMPLX(HORI,VERTI)*Z(10,5)
      Z(10,9)       =       CMPLX(-C4Q,0.0)
      Z(10,10)      =       CMPLX(C1Q,0.0)-CMPLX(SIGMA*TSQO,0.0)+
     *                      CMPLX(0.0,OMEGA*TSQO)
      RETURN
      END
C     SUBROUTINE ELILIM1
      SUBROUTINE ELILIM1(Z,K,IFM,IT)
      COMPLEX Z(10,10),ZM
      IST           =       IFM
      IL1  = IT
    1 CONTINUE
      IL2  = IL1 - 1
      IF(CABS(Z(IL1,IL1)).GT.0.0)GO TO 3
      WRITE(*,2)
    2 FORMAT(1H ,2X,'SINGULAR AT COEFFICIENT ELIMINATION')
      STOP
    3 CONTINUE
      IF(CABS(Z(IL2,IL1)).LE.0.0)GO TO 5
      ZM            =       -Z(IL2,IL1)/Z(IL1,IL1)
      DO 4 J        =       1,IL1
      Z(IL2,J)      =       Z(IL2,J) + ZM*Z(IL1,J)
    4 CONTINUE
    5 IL2           =       IL2 -1
```

```
      IF(IL2.LE.0)GO TO 6
      GO TO 3
    6 CONTINUE
      IL1          =       IL1 - 1
      IF(IL1.LT.IST)GO TO 7
      GO TO 1
    7 CONTINUE
      RETURN
      END
C     SUBROUTINE GNEXC
      SUBROUTINE GNEXC(K,GA,TA3,TA,TA2,XKEP,XKFP,TE,TF,TYPEX)
      DIMENSION GA(1),TA3(1),TA(1),TA2(1),XKEP(1),XKFP(1),TE(1),
     *TF(1)
      INTEGER TYPEX(1)
      COMMON /CX4/REXC,XEXC,RGOV,XGOV,RPSS,XPSS,FREQCY,C1DS,C1QS,C2DS,
     *C2QS,CV1,CV2,CV3,CV4,EFDO,SIGMA,OMEGA,TOL,PI,WD,FMIN,FINC,FO,
     *SBASE,Y11,Y12,Y21,Y22,C1D,C2D,C3D,C4D,C5D,C6D,C7D,C1Q,C2Q,C3Q,
     *C4Q,C5Q,C6Q,C7Q,TSDO,TFDO,TSQO,TFQO,QAXISX,DAXISX
      COMPLEX Z1,Z2,Z3,Z4,Z5,Z6,ZFWD,ZFB,ZEXC,ZONE
C      IF(TYPEX(K).EQ.4)GO TO
      IF(TYPEX(K).LE.0)GO TO 1
      ZONE         =       CMPLX(1.0,0.0)
      R1           =       GA(K)*(1.0 - SIGMA*TA3(K))
      Z1           =       CMPLX(R1,GA(K)*OMEGA*TA3(K))
      Z2           =       CMPLX(1.0 - SIGMA*TA(K),OMEGA*TA(K))
      Z3           =       CMPLX(1.0 - SIGMA*TA2(K),OMEGA*TA2(K))
      Z4           =       CMPLX(XKEP(K) - SIGMA*TE(K),OMEGA*TE(K))
      Z5           =       CMPLX(-SIGMA*XKFP(K),OMEGA*XKFP(K))
      Z6           =       CMPLX(1.0 - SIGMA*TF(K),OMEGA*TF(K))
      ZFWD         =       Z1/(Z2*Z3*Z4)
      ZFB          =       Z5/Z6
      ZEXC         =       ZFWD/(ZONE + ZFWD*ZFB)
      REXC         =       REAL(ZEXC)
      XEXC         =       AIMAG(ZEXC)
      GO TO 2
    1 REXC         =       0.0
      XEXC         =       0.0
    2 CONTINUE
```

```
      RETURN
      END
C     SUBROUTINE CCC
      SUBROUTINE CCC(NB,NG,IMAG,GB)
      INTEGER IMAG(1),GB(1)
      DO 1 I        =       1,NB
      IMAG(I)       =       0
    1 CONTINUE
      DO 2 I        =       1,NG
      J             =       GB(I)
      IMAG(J)       =       I
    2 CONTINUE
      K             =       NG + 1
      DO 5 I        =       1,NB
      IF(IMAG(I).NE.0)GO TO 4
      IMAG(I)       =       K
      K             =       K + 1
    4 CONTINUE
    5 CONTINUE
      RETURN
      END
C     SUBROUTINE YCH
      SUBROUTINE YCH(NB,PL,QL,YCHARG,YSHUNT,SB,EB,ST,ET,NL,NT,E,
     *RT,XT,TAP,PHSF,RL,XL,TYPE)
      INTEGER SB(1),EB(1),ST(1),ET(1),TYPE(1)
      DIMENSION PL(1),QL(1),YSHUNT(1),RT(1),XT(1),TAP(1),RL(1),
     *XL(1),PHSF(1)
      COMPLEX YCHARG(1),E(1),YLOAD,YY,ONE,PH
      PI            =       3.14159263
      ONE           =       CMPLX(1.0,0.0)
      DO 1 I        =       1,NB
      A             =       CABS(E(I))
      A             =       A*A
      YLOAD         =       CMPLX(PL(I),-QL(I))/CMPLX(A,0.0)
      YCHARG(I)     =       YCHARG(I) + YLOAD
      IF(TYPE(I).EQ.3)YCHARG(I)=CMPLX(1.0,0.0)
    1 CONTINUE
      DO 2 I        =       1,NL
```

```
      J              =       SB(I)
      K              =       EB(I)
      A              =       YSHUNT(I)/2.0
      IF(TYPE(J).NE.3)YCHARG(J) = YCHARG(J) + CMPLX(0.0,A)
      IF(TYPE(K).NE.3)YCHARG(K) = YCHARG(K) + CMPLX(0.0,A)
    2 CONTINUE
      DO 3 I         =       1,NT
      PHSF(I)        =       PHSF(I)*PI/180.0
      IF(ABS(TAP(I)).EQ.0.0)TAP(I)=1.0
      J              =       ST(I)
      K              =       ET(I)
      A              =       (1.0/TAP(I) - 1.0)
      B              =       A/TAP(I)
      YY             =       CMPLX(1.0,0.0)/CMPLX(RT(I),XT(I))
      IF(TYPE(J).NE.3)YCHARG(J) = YCHARG(J) + CMPLX(B,0.0)*YY
      IF(TYPE(K).NE.3)YCHARG(K) = YCHARG(K) - CMPLX(A,0.0)*YY
      N              =       NL + I
      SB(N)          =       ST(I)
      EB(N)          =       ET(I)
      RL(N)          =       RT(I)*TAP(I)
      XL(N)          =       XT(I)*TAP(I)
    3 CONTINUE
      NN             =       NL + NT
      PH             =       ONE
      DO 5 L         =       1,NN
      IF(L.GT.NL)PH=         CMPLX(COS(PHSF(L-NL)),SIN(PHSF(L-NL)))
      YY             =       CMPLX(1.0,0.0)/CMPLX(RL(L),XL(L))
      I              =       SB(L)
      J              =       EB(L)
      IF(TYPE(I).NE.3)YCHARG(I)     =    YCHARG(I) + YY/(PH*CONJG(PH))
      IF(TYPE(J).NE.3)YCHARG(J)     =    YCHARG(J) + YY
    4 CONTINUE
    5 CONTINUE
      RETURN
      END
C     SUBROUTINE GGG
      SUBROUTINE GGG(K,GW,GX,BW,BX)
      DIMENSION GW(1),GX(1),BW(1),BX(1)
```

```
      COMMON /CX4/REXC,XEXC,RGOV,XGOV,RPSS,XPSS,FREQCY,C1DS,C1QS,C2DS,
     *C2QS,CV1,CV2,CV3,CV4,EFDO,SIGMA,OMEGA,TOL,PI,WD,FMIN,FINC,FO,
     *SBASE,Y11,Y12,Y21,Y22,C1D,C2D,C3D,C4D,C5D,C6D,C7D,C1Q,C2Q,C3Q,
     *C4Q,C5Q,C6Q,C7Q,TSDO,TFDO,TSQO,TFQO,QAXISX,DAXISX
      GW(K)          =       Y11
      BW(K)          =       Y12
      BX(K)          =       Y21
      GX(K)          =       Y22
      RETURN
      END
C     SUBROUTINE RIGH
      SUBROUTINE RIGH(NB,NG,IDRIVE,Z,RH,WD)
      COMPLEX Z(10,10),RH(1)
      N              =       (NG + NB)*2
      DO 1 I         =       1,N
    1 RH(I)          =       CMPLX(0.0,0.0)
      I1             =       IDRIVE*2 - 1 + NB*2
      I2             =       I1 + 1
      RH(I1)         =       -Z(3,5)*CMPLX(WD,0.0)
      RH(I2)         =       -Z(4,5)*CMPLX(WD,0.0)
      RETURN
      END
C     SUBROUTINE SPEED
      SUBROUTINE SPEED(Z,K,SPC)
      COMPLEX Z(10,10),SPC(30,4)
      SPC(K,1)       =       -Z(5,1)/Z(5,5)
      SPC(K,2)       =       -Z(5,2)/Z(5,5)
      SPC(K,3)       =       -Z(5,3)/Z(5,5)
      SPC(K,4)       =       -Z(5,4)/Z(5,5)
      RETURN
      END
C     SUBROUTINE TORQUE
      SUBROUTINE TORQUE(TC,Z)
      COMPLEX TC(1),Z(10,10)
      TC(1)          =       Z(5,1)
      TC(2)          =       Z(5,2)
      TC(3)          =       Z(5,3)
      TC(4)          =       Z(5,4)
```

```
      TC(5)          =        Z(5,5)
      RETURN
      END
C     SUBROUTINE SPEED2
      SUBROUTINE SPEED2(SPC,SPD,EE,NB,NG,IDRIVE)
      COMPLEX SPD(1),SPC(30,4),EE(1)
      NB2            =        NB*2
      DO 5 I         =        1,NG
      IF(I.EQ.IDRIVE)GO TO 4
      I1             =        I*2 - 1
      I2             =        I*2
      J1             =        NB2 + I1
      J2             =        NB2 + I2
      SPD(I)         =        SPC(I,1)*EE(I1) + SPC(I,2)*EE(I2) +
     *                        SPC(I,3)*EE(J1) + SPC(I,4)*EE(J2)
    4 CONTINUE
    5 CONTINUE
      RETURN
      END
C     SUBROUTINE CHECK
      SUBROUTINE CHECK(NG,SPD,SIGMA,OMEGA,WD,IDRIVE,HC,ALPHA,BETA,
     *TXV,TXH,WK)
      COMPLEX SPD(1)
      DIMENSION HC(1)
      SPD(IDRIVE)    =        CMPLX(WD,0.0)
      SUM1           =        0.0
      SUM2           =        0.0
      DO 3 I         =        1,NG
      SUM1 = SUM1 + HC(I)*REAL(SPD(I))*AIMAG(SPD(I))
      SUM2 = SUM2 + HC(I)*(REAL(SPD(I))**2 - AIMAG(SPD(I))**2)
    3 CONTINUE
      SUM1           =        -SUM1*2.0
      OT2            =        ATAN2(SUM1,SUM2)
      OT             =        OT2/2.0
      WK             =        0.0
      DO 4 I         =        1,NG
      A              =        REAL(SPD(I))*COS(OT)
      B              =        AIMAG(SPD(I))*SIN(OT)
```

```
      WK          =       WK + HC(I)*(A - B)**2
    4 CONTINUE
      T1 = WD*(OMEGA*TXH - SIGMA*TXV)/(4.0*OMEGA*WK)
      T2 = WD*(OMEGA*TXV + SIGMA*TXH)/(2.0*WK)
      ALPHA       =       SIGMA + T1
      B2          =       OMEGA*OMEGA - T2 - T1*T1
      BETA        =       SQRT(B2)
      RETURN
      END
C     SUBROUTINE TT2
      SUBROUTINE TT2(NB,TC,IDRIVE,EE,TXV,TXH,WD)
      COMPLEX EE(1),TC(1),TT
      I           =       IDRIVE*2 - 1
      J           =       IDRIVE*2
      I1          =       I + NB*2
      J1          =       J + NB*2
      TT = TC(1)*EE(I) + TC(2)*EE(J) + TC(3)*EE(I1) +
     +     TC(4)*EE(J1) + CMPLX(WD,0.0)*TC(5)
      TXH         =       REAL(TT)
      TXV         =       AIMAG(TT)
      RETURN
      END
C     SUBROUTINE ARM1
      SUBROUTINE ARM1(NB,NL,NT,NG,SB,EB,ST,ET,RL,XS,RT,XT,TAP,PHSF,
     *YCHARG,ASELF,AMUT,GW,BW,GX,BX,LIST,NEXT,FAR,IMAG,GB,TYPE,LAST)
      DIMENSION RL(1),XS(1),RT(1),TAP(1),GW(1),BW(1),
     *GX(1),BX(1),PHSF(1)
      INTEGER SB(1),EB(1),ST(1),ET(1),LIST(1),NEXT(1),FAR(1),IMAG(1),
     *GB(1),TYPE(1),ENDA,ENDB
      COMPLEX YCHARG(1),ASELF(1),AMUT(1),YY,PH,ONE
      ONE         =       CMPLX(1.0,0.0)
      NN          =       (NB + NG)*2
      NB2         =       NB*2
      DO 2 N      =       1,NB
      LAST        =       LAST + 1
      ENDA        =       LAST*2 - 1
      ENDB        =       LAST*2
      I           =       IMAG(N)
```

```
      NODEA          =       I*2 - 1
      NODEB          =       I*2
      ASELF(NODEA)   =       CMPLX(REAL(YCHARG(N)),0.0)
      ASELF(NODEB)   =       CMPLX(REAL(YCHARG(N)),0.0)
      NEXT(ENDA)     =       LIST(NODEA)
      LIST(NODEA)    =       ENDA
      FAR(ENDA)      =       NODEB
      NEXT(ENDB)     =       LIST(NODEB)
      LIST(NODEB)    =       ENDB
      FAR(ENDB)      =       NODEA
      A              =       -AIMAG(YCHARG(N))
      AMUT(ENDA)     =       CMPLX(A,0.0)
      AMUT(ENDB)     =       -AMUT(ENDA)
    2 CONTINUE
      DO 3 I         =       1,NG
      I1             =       I*2 - 1
      I2             =       I*2
      ASELF(I1)      =       ASELF(I1) + GW(I)
      ASELF(I2)      =       ASELF(I2) + GX(I)
      J1             =       LIST(I1)
      J2             =       LIST(I2)
      AMUT(J1)       =       AMUT(J1) + BW(I)
      AMUT(J2)       =       AMUT(J2) + BX(I)
    3 CONTINUE
      PH             =       ONE
      DO 5 L         =       1,NL + NT
      IF(TYPE(SB(L)).EQ.3.OR.TYPE(EB(L)).EQ.3)GO TO 4
      IF(L.GT.NL)PH=         CMPLX(COS(PHSF(L-NL)),SIN(PHSF(L-NL)))
      I              =       IMAG(SB(L))
      J              =       IMAG(EB(L))
      I1             =       I*2 - 1
      I2             =       I*2
      J1             =       J*2 - 1
      J2             =       J*2
      YY             =       CMPLX(1.0,0.0)/CMPLX(RL(L),XS(L))
      YY             =       YY/CONJG(PH)
      G              =       -REAL(YY)
      B              =       -AIMAG(YY)
```

```
      BB              =       B
      IF(L.GT.NL.AND.ABS(PHSF(L-NL)).GT.0.0)BB=       -B
C     I1 & J1
      LAST            =       LAST + 1
      ENDA            =       LAST*2 - 1
      ENDB            =       LAST*2
      NODEA           =       I1
      NODEB           =       J1
      NEXT(ENDA)      =       LIST(NODEA)
      LIST(NODEA)     =       ENDA
      FAR(ENDA)       =       NODEB
      NEXT(ENDB)      =       LIST(NODEB)
      LIST(NODEB)     =       ENDB
      FAR(ENDB)       =       NODEA
      AMUT(ENDA)      =       CMPLX(G,0.0)
      AMUT(ENDB)      =       CMPLX(G,0.0)
C     I1 & J2
      LAST            =       LAST + 1
      ENDA            =       LAST*2 - 1
      ENDB            =       LAST*2
      NODEA           =       I1
      NODEB           =       J2
      NEXT(ENDA)      =       LIST(NODEA)
      LIST(NODEA)     =       ENDA
      FAR(ENDA)       =       NODEB
      NEXT(ENDB)      =       LIST(NODEB)
      LIST(NODEB)     =       ENDB
      FAR(ENDB)       =       NODEA
      AMUT(ENDA)      =       CMPLX(-B,0.0)
      AMUT(ENDB)      =       CMPLX(BB,0.0)
C     I2 & J1
      LAST            =       LAST + 1
      ENDA            =       LAST*2 - 1
      ENDB            =       LAST*2
      NODEA           =       I2
      NODEB           =       J1
      NEXT(ENDA)      =       LIST(NODEA)
      LIST(NODEA)     =       ENDA
```

```
      FAR(ENDA)     =       NODEB
      NEXT(ENDB)    =       LIST(NODEB)
      LIST(NODEB)   =       ENDB
      FAR(ENDB)     =       NODEA
      AMUT(ENDA)    =       CMPLX(B,0.0)
      AMUT(ENDB)    =       CMPLX(-BB,0.0)
C     I2 & J2
      LAST          =       LAST + 1
      ENDA          =       LAST*2 - 1
      ENDB          =       LAST*2
      NODEA         =       I2
      NODEB         =       J2
      NEXT(ENDA)    =       LIST(NODEA)
      LIST(NODEA)   =       ENDA
      FAR(ENDA)     =       NODEB
      NEXT(ENDB)    =       LIST(NODEB)
      LIST(NODEB)   =       ENDB
      FAR(ENDB)     =       NODEA
      AMUT(ENDA)    =       CMPLX(G,0.0)
      AMUT(ENDB)    =       CMPLX(G,0.0)
    4 CONTINUE
    5 CONTINUE
      RETURN
      END
C     SUBROUTINE ARM2
      SUBROUTINE ARM2(K,NB,LIST,NEXT,FAR,ASELF,AMUT,GW,GX,BW,BX,
     *LAST,Z)
      INTEGER LIST(1),NEXT(1),FAR(1),ENDA,ENDB
      DIMENSION GW(1),GX(1),BX(1),BW(1)
      COMPLEX AMUT(1),ASELF(1),Z(10,10)
      NB2           =       NB*2
      LAST          =       LAST + 1
      ENDA          =       LAST*2 - 1
      ENDB          =       LAST*2
      NODEA         =       NB2 + K*2 - 1
      NODEB         =       K*2 + NB2
      ASELF(NODEA)  =       Z(3,3)
      ASELF(NODEB)  =       Z(4,4)
```

```
      NEXT(ENDA)    =    LIST(NODEA)
      LIST(NODEA)   =    ENDA
      FAR(ENDA)     =    NODEB
      NEXT(ENDB)    =    LIST(NODEB)
      LIST(NODEB)   =    ENDB
      FAR(ENDB)     =    NODEA
      AMUT(ENDA)    =    Z(3,4)
      AMUT(ENDB)    =    Z(4,3)
      I1            =    K*2 - 1
      I2            =    K*2
      J1            =    K*2 - 1 + NB2
      J2            =    K*2 + NB2
C     I1 & J1
      LAST          =    LAST + 1
      ENDA          =    LAST*2 - 1
      ENDB          =    LAST*2
      NODEA         =    I1
      NODEB         =    J1
      NEXT(ENDA)    =    LIST(NODEA)
      LIST(NODEA)   =    ENDA
      FAR(ENDA)     =    NODEB
      NEXT(ENDB)    =    LIST(NODEB)
      LIST(NODEB)   =    ENDB
      FAR(ENDB)     =    NODEA
      AMUT(ENDA)    =    CMPLX(-GW(K),0.0)
      AMUT(ENDB)    =    Z(3,1)
C     I1 & J2
      LAST          =    LAST + 1
      ENDA          =    LAST*2 - 1
      ENDB          =    LAST*2
      NODEA         =    I1
      NODEB         =    J2
      NEXT(ENDA)    =    LIST(NODEA)
      LIST(NODEA)   =    ENDA
      FAR(ENDA)     =    NODEB
      NEXT(ENDB)    =    LIST(NODEB)
      LIST(NODEB)   =    ENDB
      FAR(ENDB)     =    NODEA
```

```
      AMUT(ENDA)    =       CMPLX(-BW(K),0.0)
      AMUT(ENDB)    =       Z(4,1)
C     I2 & J1
      LAST          =       LAST + 1
      ENDA          =       LAST*2 - 1
      ENDB          =       LAST*2
      NODEA         =       I2
      NODEB         =       J1
      NEXT(ENDA)    =       LIST(NODEA)
      LIST(NODEA)   =       ENDA
      FAR(ENDA)     =       NODEB
      NEXT(ENDB)    =       LIST(NODEB)
      LIST(NODEB)   =       ENDB
      FAR(ENDB)     =       NODEA
      AMUT(ENDA)    =       CMPLX(-BX(K),0.0)
      AMUT(ENDB)    =       Z(3,2)
C     I2 & J2
      LAST          =       LAST + 1
      ENDA          =       LAST*2 - 1
      ENDB          =       LAST*2
      NODEA         =       I2
      NODEB         =       J2
      NEXT(ENDA)    =       LIST(NODEA)
      LIST(NODEA)   =       ENDA
      FAR(ENDA)     =       NODEB
      NEXT(ENDB)    =       LIST(NODEB)
      LIST(NODEB)   =       ENDB
      FAR(ENDB)     =       NODEA
      AMUT(ENDA)    =       CMPLX(-GX(K),0.0)
      AMUT(ENDB)    =       Z(4,2)
      RETURN
      END
C     SUBROUTINE ELIMIN
      SUBROUTINE ELIMIN(CBUS,EBUS,YSELF,YMUT,LIST,NEXT,FAR,NBUS,LAST,
     *NCONN)
      COMPLEX CBUS(1),EBUS(1),YSELF(1),YMUT(1),YMULTI
      INTEGER LIST(1),NEXT(1),FAR(1),NCONN(1),END1,END2,END3,END4,END5,
     *ENDN,ENDM,ENDI,ENDJ
```

```
      DO 1 J        =       1,NBUS
      EBUS(J)       =       0.0
    1 CONTINUE
      DO 4 I        =       1,NBUS
      NCONN(I)      =       1
      IF(LIST(I).EQ.0)GO TO 3
      ENDI          =       LIST(I)
    2 CONTINUE
      NCONN(I)      =       NCONN(I) + 1
      IF(NEXT(ENDI).EQ.0)GO TO 3
      ENDI          =       NEXT(ENDI)
      GO TO 2
    3 CONTINUE
    4 CONTINUE
    5 CONTINUE
      NMAX          =       1000
      NODEM         =       0
      DO 7 I        =       1,NBUS
      IF(NCONN(I).EQ.0)GO TO 6
      IF((LIST(I).EQ.0).OR.(NCONN(I).GT.NMAX))GO TO 6
      NMAX          =       NCONN(I)
      NODEM         =       I
    6 CONTINUE
    7 CONTINUE
      IF(NODEM.EQ.0)GO TO 14
      END1          =       LIST(NODEM)
    8 NODE1         =       FAR(END1)
      L             =       (END1 + 1)/2
      I             =       L*2 - 1
      J             =       L*2
      NCONN(NODEM)  =       0
      IF(I.EQ.END1)ENDJ = J
      IF(J.EQ.END1)ENDJ = I
      IF(ABS(YMUT(ENDJ)).LE.1.0E-7)GO TO 13
      YMULTI        =       -YMUT(ENDJ)/YSELF(NODEM)
      NCONN(NODE1)  =       NCONN(NODE1) - 1
      CBUS(NODE1)   =       CBUS(NODE1) + YMULTI*CBUS(NODEM)
      YSELF(NODE1)  =       YSELF(NODE1) + YMULTI*YMUT(END1)
```

```
      YMUT(ENDJ)    =       0.0
      END2          =       LIST(NODEM)
    9 NODE2         =       FAR(END2)
      IF(NODE2.EQ.NODE1)GO TO 12
      IF(ABS(YMUT(END2)).LE.1.0E-7)GO TO 12
      END3          =       LIST(NODE1)
   10 NODE3         =       FAR(END3)
      IF(NODE3.NE.NODE2)GO TO 11
      YMUT(END3)    =       YMUT(END3) + YMULTI*YMUT(END2)
      GO TO 12
   11 END3          =       NEXT(END3)
      IF(END3.NE.0)GO TO 10
      LAST          =       LAST + 1
      END4          =       LAST*2 - 1
      END5          =       LAST*2
      NEXT(END4)    =       LIST(NODE2)
      LIST(NODE2)   =       END4
      FAR(END4)     =       NODE1
      NEXT(END5)    =       LIST(NODE1)
      LIST(NODE1)   =       END5
      FAR(END5)     =       NODE2
      YMUT(END5)    =       YMULTI*YMUT(END2)
      NCONN(NODE1)  =       NCONN(NODE1) + 1
      NCONN(NODE2)  =       NCONN(NODE2) + 1
      YMUT(END4)    =       0.0
   12 END2          =       NEXT(END2)
      IF(END2.NE.0)GO TO 9
   13 END1          =       NEXT(END1)
      IF(END1.NE.0)GO TO 8
      GO TO 5
   14 CONTINUE
      DO 16 I       =       1,NBUS
      IF((LIST(I).EQ.0).OR.(CABS(YSELF(I)).LE.1.0E-7))GO TO 15
      EBUS(I)       =       CBUS(I)/YSELF(I)
   15 LIST(I)       =       0
   16 CONTINUE
      RETURN
      END
```

```
C     SUBROUTINE FIRST
      SUBROUTINE FIRST(PI,P,PL,Q,QL,YSHUNT,VB,ANG,SBASE,E,
     *NB,NL,FO,SIGMA,OMEGA)
      DIMENSION P(1),PL(1),Q(1),QL(1),YSHUNT(1),VB(1),ANG(1)
      COMPLEX E(1)
      PI          =     3.141592653
      DO 1 J      =     1,NB
      ANG(J)      =     ANG(J)*PI/180.0
      A           =     VB(J)*COS(ANG(J))
      B           =     VB(J)*SIN(ANG(J))
      E(J)        =     CMPLX(A,B)
      P(J)        =     P(J)/SBASE
      PL(J)       =     PL(J)/SBASE
      Q(J)        =     Q(J)/SBASE
      QL(J)       =     QL(J)/SBASE
    1 CONTINUE
      DO 2 I      =     1,NL
      YSHUNT(I)   =     YSHUNT(I)/SBASE
    2 CONTINUE
      SIGMA       =     0.0
      OMEGA       =     PI*FO*2.0
      RETURN
      END
C     SUBROUTINE GXPSS
      SUBROUTINE GXPSS(K,IPSS,XG1,XT1,XT3,XT4,XT5,XT6,XT7,XT8,
     *TR,VL1,VL2)
      DIMENSION XG1(1),XT1(1),XT3(1),XT4(1),XT5(1),XT6(1),XT7(1),
     *XT8(1),TR(1),VL1(1),VL2(1)
      INTEGER IPSS(1)
      COMPLEX ZT1,ZT3,ZT4,ZT5,ZT6,ZT7,ZT8,ZTR,ZPSS
      COMMON /CX4/REXC,XEXC,RGOV,XGOV,RPSS,XPSS,FREQCY,C1DS,C1QS,C2DS,
     *C2QS,CV1,CV2,CV3,CV4,EFDO,SIGMA,OMEGA,TOL,PI,WD,FMIN,FINC,FO,
     *SBASE,Y11,Y12,Y21,Y22,C1D,C2D,C3D,C4D,C5D,C6D,C7D,C1Q,C2Q,C3Q,
     *C4Q,C5Q,C6Q,C7Q,TSDO,TFDO,TSQO,TFQO,QAXISX,DAXISX
      IF(IPSS(K).NE.0)GO TO 1
      RPSS        =     0.0
      XPSS        =     0.0
      GO TO 2
```

```
    1 CONTINUE
      ZT1          =       CMPLX(XG1(K),0.0)
      ZT1          =       ZT1/CMPLX(1.0-SIGMA*XT1(K),OMEGA*XT1(K))
      ZT3          =       CMPLX(1.0-SIGMA*XT3(K),OMEGA*XT3(K))
      ZT4          =       CMPLX(1.0-SIGMA*XT4(K),OMEGA*XT4(K))
      ZT5          =       CMPLX(1.0-SIGMA*XT5(K),OMEGA*XT5(K))
      ZT6          =       CMPLX(-SIGMA*XT6(K),OMEGA*XT6(K))
      IF(XT6(K).LE.0.0)ZT6 = CMPLX(1.0,0.0)
      ZT7          =       CMPLX(1.0-SIGMA*XT7(K),OMEGA*XT7(K))
      ZT8          =       CMPLX(1.0-SIGMA*XT8(K),OMEGA*XT8(K))
      ZPSS         =       ZT1*ZT6*ZT7*ZT8/(ZT3*ZT4*ZT5)
      ZTR          =       CMPLX(1.0-SIGMA*TR(K),OMEGA*TR(K))
      ZPSS         =       ZTR*ZPSS
      RPSS         =       REAL(ZPSS)
      XPSS         =       AIMAG(ZPSS)
    2 CONTINUE
      RETURN
      END
C     SUBROUTINE GXGOV
      SUBROUTINE GXGOV(K,IGOV,G1,G2,G3,G4,T1,T2,T3,T4,T5,T6)
      INTEGER IGOV(1)
      DIMENSION G1(1),G2(1),G3(1),G4(1),T1(1),T2(1),T3(1),T4(1),T5(1),
     *T6(1)
      COMPLEX Z1,Z2,Z3,Z4,Z5,Z6,ZC,ZHP
      COMMON /CX4/REXC,XEXC,RGOV,XGOV,RPSS,XPSS,FREQCY,C1DS,C1QS,C2DS,
     *C2QS,CV1,CV2,CV3,CV4,EFDO,SIGMA,OMEGA,TOL,PI,WD,FMIN,FINC,FO,
     *SBASE,Y11,Y12,Y21,Y22,C1D,C2D,C3D,C4D,C5D,C6D,C7D,C1Q,C2Q,C3Q,
     *C4Q,C5Q,C6Q,C7Q,TSDO,TFDO,TSQO,TFQO,QAXISX,DAXISX
      IF(IGOV(K).NE.0)GO TO 1
      RGOV         =       0.0
      XGOV         =       0.0
      GO TO 2
    1 CONTINUE
      Z1           =       CMPLX(1.0-SIGMA*T1(K),OMEGA*T1(K))
      R2           =       G1(K)*(1.0-SIGMA*T2(K))
      Z2           =       CMPLX(R2,G1(K)*T2(K)*OMEGA)
      Z3           =       CMPLX(1.0-SIGMA*T3(K),OMEGA*T3(K))
      Z4           =       CMPLX(1.0-SIGMA*T4(K),OMEGA*T4(K))
```

```
      Z5          =     CMPLX(1.0-SIGMA*T5(K),OMEGA*T5(K))
      Z6          =     CMPLX(1.0-SIGMA*T6(K),OMEGA*T6(K))
      ZC          =     Z2/(Z1*Z3*Z4)
      ZHP         =     CMPLX(1.0-G3(K),0.0)
      A           =     1.0 -G4(K)
      ZHP         =     ZHP+CMPLX(G3(K)*A,0.0)/Z6
      ZHP         =     ZHP*CMPLX(G2(K),0.0)/Z5
      ZHP         =     ZHP+CMPLX(1.0-G2(K),0.0)
      ZHP         =     ZHP*ZC
      RGOV        =     REAL(ZHP)
      XGOV        =     AIMAG(ZHP)
    2 CONTINUE
      RETURN
      END
```

ประวัติผู้เขียน

นาย วิริยะ เจริญศิลป์ เป็นบุตรของ นาย วิยุทธ์ และ นาง วันเพ็ญ เจริญศิลป์ เกิดที่ หมู่ที่ 1 ตำบลท่างิ้ว อำเภอบรรพตพิสัย จังหวัดนครสวรรค์ เริ่มเข้าการศึกษาระดับประถมศึกษาตอนต้นที่โรงเรียนวัดท่างิ้ว อำเภอบรรพตพิสัย จังหวัดนครสวรรค์ เมื่อจบการศึกษาระดับประถมศึกษาตอนต้นแล้ว ได้ศึกษาต่อในระดับประถมศึกษาตอนปลายที่โรงเรียนวัดส้มเสี้ยว อำเภอบรรพตพิสัย จังหวัดนครสวรรค์ เมื่อจบการศึกษาระดับประถมศึกษาตอนปลายแล้วได้ศึกษาต่อในระดับมัธยมศึกษาตอนต้นและมัธยมศึกษาตอนปลายที่โรงเรียนนครสวรรค์ อำเภอเมือง จังหวัดนครสวรรค์ เข้ารับการศึกษาระดับอุดมศึกษาที่ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้า วิทยาเขต เจ้าคุณทหาร-ลาดกระบัง เมื่อปีการศึกษา 2521 ในสาขาวิศวกรรมไฟฟ้า จบการศึกษาระดับอุดมศึกษาเมื่อปีการศึกษา 2525 เข้าเรียนในบัณฑิตวิทยาลัย จุฬาลงกรณ์มหาวิทยาลัย ในปีการศึกษา 2528